aws

Guia do usuário

Amazon Linux 2023

Amazon Linux 2023: Guia do usuário

Table of Contents

O que é o Amazon Linux 2023?

O Amazon Linux 2023 (AL2023) é a próxima geração do Amazon Linux da Amazon Web Services (AWS). Com o AL2 023, você pode desenvolver e executar aplicativos corporativos e em nuvem em um ambiente de execução seguro, estável e de alto desempenho. Além disso, você obtém um ambiente de aplicativos que oferece suporte de longo prazo com acesso às mais recentes inovações no Linux. AL2023 é fornecido sem custo adicional.

AL2023 é o sucessor do Amazon Linux (2)AL2. Para obter informações sobre as diferenças entre AL2 023 e AL2, consulte Comparando AL2 e AL2 023 e Package changes in AL2 023.

Tópicos

Cadência de lançamento

O Amazon Linux 2023 (AL2023) foi lançado em março de 2023 e terá suporte até 30 de junho de 2029. Há duas fases de suporte:

- Suporte padrão — Durante essa fase, o lançamento recebe atualizações trimestrais de versões secundárias. A fase de suporte padrão termina em 30 de junho de 2027.
- Manutenção — Durante essa fase, a versão recebe somente atualizações de segurança e correções críticas de bugs. Essas atualizações são publicadas assim que estão disponíveis. A fase de manutenção termina em 30 de junho de 2029.

Liberações principais e secundárias

A cada lançamento do Amazon Linux (versão principal, versão secundária ou lançamento de segurança), lançamos uma nova imagem de máquina da Amazon (AMI) Linux.

- Versão principal: inclui novos recursos e melhorias em segurança e desempenho em toda a pilha. As melhorias podem incluir grandes mudanças no kernel, no conjunto de ferramentas, Glib C, OpenSSLe quaisquer outras bibliotecas e utilitários do sistema. Os principais lançamentos do Amazon Linux são baseados em parte na versão atual da distribuição upstream do Fedora Linux. AWS pode adicionar ou substituir pacotes específicos de outros upstreams que não sejam do Fedora.
- Liberação de uma versão secundária: uma atualização trimestral que inclui atualizações de segurança, correções de bugs e novos recursos e pacotes. Cada versão secundária é uma lista cumulativa de atualizações que inclui correções de segurança e bugs, além de novos recursos e pacotes. Essas versões podem incluir tempos de execução de linguagem mais recentes, como PHP. Eles também podem incluir outros pacotes de software populares, como Ansible e Docker.

Consumindo novos lançamentos

As atualizações são fornecidas por meio de uma combinação de novas versões de imagem de máquina da Amazon (AMI) e dos novos repositórios correspondentes. Por padrão, uma nova AMI e o repositório para o qual ela aponta são acoplados. No entanto, você pode direcionar suas EC2 instâncias da Amazon em execução para versões mais recentes do repositório ao longo do tempo para aplicar atualizações nas instâncias em execução. Você também pode atualizar lançando novas instâncias das mais recentes AMIs.

Política de suporte de longo prazo

O Amazon Linux fornece atualizações para todos os seus pacotes e mantém a compatibilidade em uma versão principal para seus aplicativos criados no Amazon Linux. Pacotes principais, como o glibc biblioteca, OpenSSL, OpenSSH, e o DNF O gerenciador de pacotes recebe suporte durante toda a vida útil da versão AL2 023 principal. Pacotes que não fazem parte dos pacotes principais são compatíveis com base em suas fontes upstream específicas. Você pode ver o status de suporte específico e as datas de pacotes individuais executando o comando a seguir.

```
$ sudo dnf supportinfo --pkg packagename
```

Você pode obter informações sobre todos os pacotes atualmente instalados executando o comando a seguir.

```
$ sudo dnf supportinfo --show installed
```

A lista completa dos pacotes principais é finalizada durante a pré-visualização. Se você quiser ver mais pacotes incluídos como pacotes principais, conte-nos. Avaliamos à medida que coletamos feedback. O feedback sobre AL2 023 pode ser fornecido por meio de seu AWS representante designado ou registrando um problema no repositório amazon-linux-2023 em. GitHub

Nomeação e controle de versão

AL2O 023 fornece uma versão menor a cada três meses durante os dois anos de suporte padrão. Cada versão é identificada por um incremento de 0 a N. 0 se refere à versão principal original dessa iteração. Todos os lançamentos serão chamados de Amazon Linux 2023. Quando a próxima versão do Amazon Linux for lançada, o AL2 023 entrará no suporte estendido e receberá atualizações para atualizações de segurança e correções de bugs críticos.

Por exemplo, versões menores de AL2 023 têm o seguinte formato:

- `2023.0.20230301`
- `2023.1.20230601`
- `2023.2.20230901`

O AL2 023 correspondente AMIs tem o seguinte formato:

- `al2023-ami-2023.0.20230301.0-kernel-6.1-x86_64`
- `al2023-ami-2023.1.20230601.0-kernel-6.1-x86_64`
- `al2023-ami-2023.2.20230901.0-kernel-6.1-x86_64`

Em uma versão secundária específica, os lançamentos regulares da AMI ocorrem com um timestamp da data do lançamento da AMI.

- `al2023-ami-2023.0.20230301.0-kernel-6.1-x86_64`
- `al2023-ami-2023.0.20230410.0-kernel-6.1-x86_64`
- `al2023-ami-2023.0.20230520.0-kernel-6.1-x86_64`

O método recomendado para identificar uma instância AL2 ou AL2 023 começa com a leitura da string Common Platform Enumeration (CPE) de. `/etc/system-release-cpe` Em seguida, divida a string em seus campos. Por fim, leia os valores da plataforma e da versão.

AL2O 023 também introduz novos arquivos para identificação da plataforma:

- `/etc/amazon-linux-release` symlinks para `/etc/system-release`
- `/etc/amazon-linux-release-cpe` symlinks para `/etc/system-release-cpe`

Esses dois arquivos indicam que uma instância é Amazon Linux. Não é necessário ler um arquivo ou dividir a string em campos, a menos que você queira saber os valores específicos da plataforma e da versão.

Otimizações operacionais e de desempenho

Kernel Amazon Linux 6.1

- AL2O 023 usa os drivers mais recentes para dispositivos Elastic Network Adapter (ENA) e Elastic Fabric Adapter (EFA). AL2O 023 se concentra em backports de desempenho e funcionalidade para hardware na infraestrutura da Amazon EC2 .
- O kernel live patching está disponível para os tipos de instância `x86_64` e `aarch64`. Isso reduz a necessidade de reinicializar com frequência.
- Todas as configurações de compilação e tempo de execução do kernel incluem muitas das mesmas otimizações operacionais e de desempenho do. AL2

Seleção do conjunto de ferramentas básico e sinalizadores de construção padrão

- AL2Os pacotes 023 são construídos com otimizações de compilador () `-O2` habilitadas por padrão
- AL2Os pacotes 023 são criados exigindo `x86-64v2` `x86-64` sistemas (`-march=x86-64-v2`) e Graviton 2 ou superior para `aarch64` ()`-march=armv8.2-a+crypto -mtune=neoverse-n1`.
- AL2Os pacotes 023 são construídos com a vetorização automática ativada (). `-ftree-vectorize`
- AL2Os pacotes 023 são criados com o Link Time Optimization (LTO) ativado.
- AL2023 usa as versões atualizadas do Rust, Clang/LLVM e Go.

Seleção e versões de pacotes

- Alguns backports para os principais componentes do sistema incluem várias melhorias de desempenho para execução na EC2 infraestrutura da Amazon, especialmente nas instâncias do Graviton.
- AL2O 023 é integrado com vários Serviços da AWS recursos. Isso inclui o AWS CLI SSM Agent, o Amazon Kinesis Agent e. CloudFormation
- AL2023 usa o Amazon Corretto como o Java Development Kit (JDK).
- AL2O 023 fornece mecanismos de banco de dados e atualizações de tempo de execução da linguagem de programação para versões mais recentes à medida que são lançadas por projetos upstream. Os tempos de execução da linguagem de programação com novas versões são adicionados quando são lançados.

Implantação em um ambiente de nuvem

- A AMI básica AL2 023 e as imagens de contêiner são atualizadas com frequência para oferecer suporte à substituição de instâncias de patches.
- As atualizações do kernel estão incluídas nas atualizações AL2 023 da AMI. Isso significa que você não precisa usar comandos como `yum update` e `reboot` para atualizar seu kernel.
- Além da AMI AL2 023 padrão, uma AMI mínima e uma imagem de contêiner também estão disponíveis. Escolha a AMI mínima para executar um ambiente com o número mínimo de pacotes necessários para executar seu serviço.
- Por padrão, AL2 023 AMIs e os contêineres estão bloqueados em uma versão específica dos repositórios de pacotes. Não há atualização automática quando eles são lançados. Isso significa que você está sempre no controle de quando ingere qualquer atualização de pacote. Você sempre pode testar em um ambiente beta/gama antes de começar a produção. Se houver algum problema, você pode usar o caminho de reversão pré-validado.

Relacionamento com o Fedora

AL2O 023 mantém seus próprios ciclos de vida de lançamento e suporte independentes do Fedora. AL2O 023 fornece versões atualizadas de software de código aberto, uma variedade maior de pacotes e lançamentos frequentes. Isso preserva os conhecidos sistemas operacionais baseados em RPM.

A versão Generally Available (GA) do AL2 023 não é diretamente comparável a nenhuma versão específica do Fedora. A versão AL2 023 GA inclui componentes do Fedora 34, 35 e 36. Alguns dos componentes são iguais aos do Fedora e alguns são modificados. Outros componentes se assemelham mais aos componentes do CentOS Stream 9 ou foram desenvolvidos de forma independente. O kernel Amazon Linux é originado das opções de suporte de longo prazo que estão no kernel.org, escolhidas independentemente do Fedora.

Personalizado cloud-init

A ferramenta cloud-init O pacote é um aplicativo de código aberto que inicializa imagens Linux em um ambiente de computação em nuvem. Para obter mais informações, consulte a documentação do cloud-init.

AL2023 contém uma versão personalizada do cloud-init. Com cloud-init, você pode especificar o que ocorre com sua instância no momento da inicialização.

Ao iniciar uma instância, você pode usar os campos de dados do usuário para transmitir ações para cloud-init. Isso significa que você pode usar Amazon Machine Images (AMIs) comuns para vários casos de uso e configurá-los dinamicamente ao iniciar uma instância. AL2023 também usa cloud-init para configurar a `ec2-user` conta.

AL2023 usa o cloud-init ações em `/etc/cloud/cloud.cfg.d` `/etc/cloud/cloud.cfg` e. Você pode criar seu próprio cloud-init arquivos de ação no `/etc/cloud/cloud.cfg.d` diretório. Cloud-init lê todos os arquivos desse diretório em ordem lexicográfica. Arquivos mais recentes substituem arquivos mais antigos. Quando cloud-init inicia uma instância, a cloud-init o pacote executa as seguintes tarefas de configuração:

- Definir o local padrão
- Define o nome do host
- Analisa e manipula dados do usuário
- Gerenciar chaves SSH privadas de host.
- Adicionar as chaves SSH públicas de um usuário ao `.ssh/authorized_keys` para facilitar login e administração.
- Prepara os repositórios para gerenciamento de pacotes.
- Lidar com as ações de pacotes definidas nos dados do usuário.
- Executa scripts de usuário que estão nos dados do usuário

- Monta volumes de armazenamento de instâncias, se aplicável
  - Por padrão, se o volume de armazenamento de instâncias `ephemeral0` estiver presente e contiver um sistema de arquivos válido, o volume de armazenamento de instâncias será montado em `/media/ephemeral0`. Caso contrário, ele não está montado.
  - Por padrão, para os tipos de instância `m1.small` e `c1.medium`, todos os volumes de troca associados à instância são montados.
  - Você pode substituir a montagem padrão do volume do armazenamento de instâncias com o seguinte cloud-init diretiva:

    ```
    #cloud-config
    mounts:
    - [ ephemeral0 ]
    ```

    Para obter mais controle sobre as montagens, consulte Montagens no cloud-init documentação.
  - Quando uma instância é executada, os volumes de armazenamento de instâncias compatíveis com TRIM não são formatados. Antes de montar volumes de armazenamento de instâncias, você deve particionar e formatar volumes de armazenamento de instâncias.

    Para obter mais informações, consulte Suporte ao volume TRIM do armazenamento de instâncias no Guia do EC2 usuário da Amazon.
  - Ao iniciar suas instâncias, você pode usar o módulo `disk_setup` para particionar e formatar os volumes do seu armazenamento de instância.

    Para obter mais informações, consulte Configuração de disco no cloud-init documentação.

Para obter informações sobre o uso cloud-init com SELinux, vejaUse cloud-init para ativar o `enforcing` modo.

Para obter mais informações sobre cloud-init formatos de dados do usuário, consulte Formatos de dados do usuário no cloud-init documentação.

Atualizações e recursos de segurança

AL2O 023 fornece muitas atualizações e soluções de segurança.

Tópicos

- Gerenciar atualizações

- Segurança na nuvem
- SELinux modos
- Programa de conformidade
- Padrão do servidor SSH
- Principais características do OpenSSL 3

Gerenciar atualizações

Aplique atualizações de segurança usando DNF e versões do repositório. Para obter mais informações, consulte Gerencie atualizações de pacotes e sistemas operacionais em AL2 023.

Segurança na nuvem

A segurança é uma responsabilidade compartilhada entre você AWS e você. O modelo de responsabilidade compartilhada descreve isso como segurança na nuvem e segurança na nuvem. Para obter mais informações, consulte Segurança e compatibilidade no Amazon Linux 2023.

SELinux modos

Por padrão, SELinux está ativado e definido para o modo permissivo em AL2 023. No modo permissive, as negações de permissão são registradas, mas não aplicadas.

As SELinux políticas definem permissões para usuários, processos, programas, arquivos e dispositivos. Com SELinux, você pode escolher uma das duas políticas. As políticas são direcionadas ou de segurança multinível (MLS).

Para obter mais informações sobre SELinux modos e políticas, consulte SELinux Modos de configuração para AL2 023 e o SELinux Project Wiki.

Programa de conformidade

Auditores independentes avaliam a segurança e a conformidade do AL2 023 junto com muitos programas de AWS conformidade.

Padrão do servidor SSH

AL2O 023 inclui o OpenSSH 8.7. Por padrão, o OpenSSH 8.7 `ssh-rsa` desativa o algoritmo de troca de chaves. Para obter mais informações, consulte Configuração do servidor SSH padrão.

Principais características do OpenSSL 3

- O Certificate Management Protocol (CMP, RFC 4210) inclui CRMF (RFC 4211) e transferência HTTP (RFC 6712).
- A HTTP or HTTPS cliente em libcrypto suporta GET and POST ações, redirecionamento, simples e ASN.1-conteúdo codificado, proxies e tempos limite.
- A ferramenta EVP_KDF funciona com funções de derivação de chaves.
- A ferramenta EVP_MAC API trabalha com MACs.
- Kernel Linux TLS apoio.

Para obter mais informações, consulte o Guia de migração do OpenSSL.

Serviço de redes

O projeto de código aberto `systemd-networkd` está amplamente disponível nas distribuições Linux modernas. O projeto usa uma linguagem de configuração declarativa semelhante ao resto da estrutura `systemd`. Seus principais tipos de arquivo de configuração são os arquivos `.network` e `.link`.

O `amazon-ec2-net-utils` pacote gera configurações específicas da interface no diretório `/run/systemd/network`. Essas configurações habilitam ambas IPv4 e a IPv6 rede em interfaces quando elas estão conectadas a uma instância. Essas configurações também instalam regras de roteamento de políticas que ajudam a garantir que o tráfego de origem local seja roteado para a rede por meio da interface de rede da instância correspondente. Essas regras garantem que o tráfego correto seja roteado pela Elastic Network Interface (ENI) a partir dos endereços ou prefixos associados. Para obter mais informações sobre o uso do ENI, consulte Usando o ENI no Guia EC2 do usuário da Amazon.

Você pode personalizar esse comportamento de rede colocando um arquivo de configuração personalizado no diretório `/etc/systemd/network` para substituir as configurações padrão contidas em `/run/systemd/network`.

A documentação do systemd.network descreve como o serviço `systemd-networkd` determina a configuração que se aplica a uma interface específica. Ele também gera nomes alternativos, conhecidos como altnames, para que as interfaces apoiadas por ENI reflitam as propriedades de

vários AWS recursos. Essas propriedades de interface compatíveis com ENI são o campo `ENI ID` e `DeviceIndex` do anexo ENI. Você pode se referir a essas interfaces usando suas propriedades ao usar várias ferramentas, como o comando `ip`.

AL2023 nomes de interface de instância são gerados usando o esquema de nomenclatura de `systemd` slots. Para obter mais informações, consulte esquema nomenclatura do systemd.net.

Além disso, o AL2 023 usa o algoritmo de agendamento de transmissão de rede de gerenciamento `fq_codel` ativo de filas por padrão. Para obter mais informações, consulte a CoDelvisão geral.

Pacotes principais do conjunto de ferramentas glibc, gcc, binutils

Um subconjunto de pacotes no Amazon Linux é designado como pacotes principais da cadeia de ferramentas. Como parte importante do AL2 023, os pacotes principais recebem cinco anos de suporte. Podemos alterar a versão de um pacote, mas o suporte de longo prazo se aplica ao pacote incluído na versão do Amazon Linux.

Esses três pacotes principais fornecem uma cadeia de ferramentas do sistema que é usada para criar a maioria dos softwares na distribuição Amazon Linux.

Pacote	Definição	Finalidade
glibc 2.34	Sistema C biblioteca	Usado pela maioria dos programas binários que fornecem funções padrão e pela interface entre os programas e o kernel.
gcc 11.2	gcc suíte de compiladores	Compila C, C++, Fortran.
binutils 2.35	Assembler e vinculador, além de outras ferramentas binárias	Manipula ou inspeciona programas binários.

Recomendamos que as atualizações sejam atualizadas para qualquer glibc as bibliotecas são seguidas por uma reinicialização. Para atualizações de pacotes que controlam serviços, pode ser suficiente reiniciar os serviços para obter as atualizações. No entanto, a reinicialização do sistema garante que todas as atualizações anteriores de pacotes e bibliotecas sejam concluídas.

Ferramenta de gerenciamento de pacotes

A ferramenta padrão de gerenciamento de pacotes de software em AL2 023 é DNF. DNF é o sucessor de YUM, a ferramenta de gerenciamento de pacotes em AL2.

DNF é semelhante a YUM em seu uso. Muitas DNF comandos e opções de comando são iguais a YUM comandos. Em uma interface de linha de comandos (CLI), na maioria dos casos `dnf` substitui `yum`.

Por exemplo, para os seguintes AL2 `yum` comandos:

```
$ sudo yum install packagename
$ sudo yum search packagename
$ sudo yum remove packagename
```

Em AL2 023, eles se tornam os seguintes comandos:

```
$ sudo dnf install packagename
$ sudo dnf search packagename
$ sudo dnf remove packagename
```

Em AL2 023, o `yum` comando ainda está disponível, mas como um ponteiro para o `dnf` comando. Portanto, quando o `yum` comando é usado no shell ou em um script, todos os comandos e opções são iguais aos DNF CLI. Para obter mais informações sobre as diferenças entre YUM CLI e o DNF CLI, consulte Mudanças em DNF CLI em comparação com YUM.

Para obter uma referência completa dos comandos e opções do comando `dnf`, consulte a página do manual `man dnf`. Para obter mais informações, consulte DNF Referência de comando.

Configuração do servidor SSH padrão

Se você tem clientes SSH de vários anos atrás, talvez veja um erro ao se conectar a uma instância. Se o erro indicar que não foi encontrado nenhum tipo de chave de host correspondente, atualize sua chave de host SSH para solucionar esse problema.

Desativação padrão de assinaturas **`ssh-rsa`**

AL2O 023 inclui uma configuração padrão que desativa o algoritmo de chave de `ssh-rsa` host herdado e gera um conjunto reduzido de chaves de host. Os clientes devem oferecer suporte ao

algoritmo da chave de host `ssh-ed25519` ou `ecdsa-sha2-nistp256`ao algoritmo da chave do host.

A configuração padrão aceita qualquer um desses algoritmos de troca de chaves:

- `curve25519-sha256`
- `curve25519-sha256@libssh.org`
- `ecdh-sha2-nistp256`
- `ecdh-sha2-nistp384`
- `ecdh-sha2-nistp521`
- `diffie-hellman-group-exchange-sha256`
- `diffie-hellman-group14-sha256`
- `diffie-hellman-group16-sha512`
- `diffie-hellman-group18-sha512`

Por padrão, AL2 023 gera `ed25519` e `ECDSA` hospeda chaves. Os clientes oferecem suporte ao algoritmo da chave de host `ssh-ed25519` ou `ecdsa-sha2-nistp256`. Ao se conectar por SSH a uma instância, você deve usar um cliente que ofereça suporte a um algoritmo compatível, como `ssh-ed25519` ou `ecdsa-sha2-nistp256`. Se você precisar usar outros tipos de chave, substitua a lista de chaves geradas por um fragmento de `cloud-config` nos dados do usuário.

No exemplo a seguir, `cloud-config` gera uma chave de host `rsa` com as chaves `ecdsa` e `ed25519`.

```
#cloud-config
 ssh_genkeytypes:
 - ed25519
 - ecdsa
 - rsa
```

Se você usa um par de chaves RSA para autenticação de chave pública, seu cliente SSH deve oferecer suporte a uma assinatura `rsa-sha2-256` ou `rsa-sha2-512`. Se você estiver usando um cliente incompatível e não conseguir fazer o upgrade, reative o suporte de `ssh-rsa` na sua instância. Para reativar o `ssh-rsa` suporte, ative a política de criptografia `LEGACY` do sistema usando os seguintes comandos.

```
$ sudo dnf install crypto-policies-scripts
```

```
$ sudo update-crypto-policies --set LEGACY
```

Para obter mais informações sobre o gerenciamento de chaves de host, consulte Chaves de host do Amazon Linux.

Funcionalidade obsoleta em 2023 AL2

A funcionalidade obsoleta AL2 e não presente em AL2 023 está documentada aqui. Essa é uma funcionalidade, como recursos e pacotes, que estão presentes em AL2, mas não em AL2 023, e não serão adicionados a AL2 023. Para obter mais informações sobre por quanto tempo a funcionalidade é suportada AL2, consulte Funcionalidade obsoleta em. AL2

Também há uma funcionalidade no AL2 023 que está obsoleta e será removida em uma versão futura. Este capítulo descreve o que é essa funcionalidade, quando ela não é mais suportada e quando será removida do Amazon Linux. Compreender a funcionalidade obsoleta ajudará você a implantar o AL2 023 e a se preparar para a próxima versão principal do Amazon Linux.

Tópicos

Pacotes do **`compat-`**

Todos os pacotes AL2 com o prefixo de `compat-` são fornecidos para compatibilidade binária com binários mais antigos que ainda não foram reconstruídos para as versões modernas do pacote. Cada nova versão principal do Amazon Linux não transferirá nenhum `compat-` pacote de versões anteriores.

Todos os `compat-` pacotes em uma versão do Amazon Linux (por exemplo AL2) estão obsoletos e não estão presentes na versão subsequente (por exemplo AL2, 023). É altamente recomendável que o software seja reconstruído com base nas versões atualizadas das bibliotecas.

Funcionalidade obsoleta descontinuada em, removida em AL1 AL2

Esta seção descreve a funcionalidade que está disponível e não está mais disponível no AL2. AL1

> **Note**
>
> Como parte da fase de suporte de manutenção do AL1, alguns pacotes tinham uma data end-of-life (EOL) anterior à EOL de. AL1 Para obter mais informações, consulte Declarações de suporte AL1 do Package.

> **Note**
>
> Algumas AL1 funcionalidades foram descontinuadas em versões anteriores. Para obter informações, consulte as AL1 Notas de versão.

Tópicos

x86 de 32 bits (i686) AMIs

Como parte da versão 2014.09 do, a AL1 Amazon Linux anunciou que seria a última versão a produzir 32 bits. AMIs Portanto, a partir da versão 2015.03 do, o AL1 Amazon Linux não suporta mais a execução do sistema no modo de 32 bits. AL2 oferece suporte de tempo de execução limitado para binários de 32 bits em hosts x86-64 e não fornece pacotes de desenvolvimento para permitir a criação de novos binários de 32 bits. AL2O 023 não inclui mais nenhum pacote de espaço de usuário de 32 bits. Recomendamos que os usuários concluam a transição para o código de 64 bits antes de migrar para AL2 023.

Se você precisar executar binários de 32 bits em AL2 023, é possível usar o espaço de usuário de 32 bits de AL2 dentro de um AL2 contêiner executado sobre 023. AL2

`aws-apitools-*`substituído por AWS CLI

Antes do lançamento do AWS CLI em setembro de 2013, AWS disponibilizou um conjunto de utilitários de linha de comando, implementados em Java, que permitia que os usuários fizessem chamadas de EC2 API da Amazon. Essas ferramentas foram descontinuadas em 2015, AWS CLI

tornando-se a forma preferida de interagir com a Amazon a EC2 APIs partir da linha de comando. O conjunto de utilitários de linha de comando inclui os seguintes `aws-apitools-*` pacotes.

- `aws-apitools-as`
- `aws-apitools-cfn`
- `aws-apitools-common`
- `aws-apitools-ec2`
- `aws-apitools-elb`
- `aws-apitools-mon`

O suporte upstream para os `aws-apitools-*` pacotes terminou em março de 2017. Apesar da falta de suporte upstream, o Amazon Linux continuou a fornecer alguns desses utilitários de linha de comando, como, por exemplo`aws-apitools-ec2`, para fornecer compatibilidade com versões anteriores aos usuários. AWS CLI É uma ferramenta mais robusta e completa do que os `aws-apitools-*` pacotes, pois é mantida ativamente e fornece um meio de usar tudo AWS APIs.

Os `aws-apitools-*` pacotes foram descontinuados em março de 2017 e não receberão mais atualizações. Todos os usuários de qualquer um desses pacotes devem migrar para o o AWS CLI assim que possível. Esses pacotes não estão presentes em AL2 023.

AL1 também forneceu os `aws-apitools-rds` pacotes `aws-apitools-iam` e, que foram descontinuados e não estão presentes no AL1 Amazon Linux a partir de então. AL2

`systemd`substitui em **`upstart`** AL2

AL2 foi a primeira versão do Amazon Linux a usar o sistema `systemd` init, substituindo `upstart` in AL1. Qualquer configuração `upstart` específica deve ser alterada como parte da migração AL1 para uma versão mais recente do Amazon Linux. Não é possível usar `systemd` on AL1, portanto, a mudança de `upstart` para só `systemd` pode ser feita como parte da migração para uma versão principal mais recente do Amazon Linux, como AL2 ou AL2 023.

Funcionalidade obsoleta AL2 e removida em 023 AL2

Esta seção descreve a funcionalidade que está disponível e não está mais disponível em AL2 023. AL2

Tópicos

- Pacotes x86 (i686) de 32 bits
- aws-apitools-*substituído por AWS CLI
- awslogsdescontinuado em favor do agente Amazon Logs unificado CloudWatch
- bzrsistema de controle de revisão
- grupo v1
- log4jhotpatch () log4j-cve-2021-44228-hotpatch
- lsb_release e o pacote system-lsb-core
- mcrypt
- OpenJDK (7) java-1.7.0-openjdk
- Python 2.7
- rsyslog-opensslsubstitui rsyslog-gnutls
- Serviço de informações de rede (NIS)/yp
- Vários nomes de domínio na Amazon VPC create-dhcp-options
- Sun RPC no glibc
- Impressão digital da chave OpenSSH no registro audit
- ld.goldVinculador
- ping6

Pacotes x86 (i686) de 32 bits

Como parte da versão 2014.09 do AL1, anunciamos que seria a última versão a produzir 32 bits. AMIs Portanto, a partir da versão 2015.03 do, o AL1 Amazon Linux não suporta mais a execução do sistema no modo de 32 bits. AL2 fornece suporte de tempo de execução limitado para binários de 32 bits em hosts x86-64 e não fornece pacotes de desenvolvimento para permitir a criação de novos binários de 32 bits. AL2O 023 não inclui mais nenhum pacote de espaço de usuário de 32 bits. Recomendamos que os clientes concluam a transição para o código de 64 bits.

Se você precisar executar binários de 32 bits em AL2 023, é possível usar o espaço de usuário de 32 bits de AL2 dentro de um AL2 contêiner executado sobre 023. AL2

aws-apitools-*substituído por AWS CLI

Antes do lançamento do AWS CLI em setembro de 2013, AWS disponibilizou um conjunto de utilitários de linha de comando, implementados emJava, que permitiam aos clientes fazer chamadas

de EC2 API da Amazon. Essas ferramentas foram descontinuadas em 2015, AWS CLI tornando-se a forma preferida de interagir com a Amazon a EC2 APIs partir da linha de comando. Isso inclui os seguintes `aws-apitools-*` pacotes.

- `aws-apitools-as`
- `aws-apitools-cfn`
- `aws-apitools-common`
- `aws-apitools-ec2`
- `aws-apitools-elb`
- `aws-apitools-mon`

O suporte upstream para os `aws-apitools-*` pacotes terminou em março de 2017. Apesar da falta de suporte upstream, o Amazon Linux continuou a fornecer alguns desses utilitários de linha de comando (como`aws-apitools-ec2`) para oferecer compatibilidade com versões anteriores aos clientes. AWS CLI É uma ferramenta mais robusta e completa do que os `aws-apitools-*` pacotes, pois é mantida ativamente e fornece um meio de usar tudo AWS APIs.

Os `aws-apitools-*` pacotes foram descontinuados em março de 2017 e não receberão mais atualizações. Todos os usuários de qualquer um desses pacotes devem migrar para o o AWS CLI assim que possível. Esses pacotes não estão presentes em AL2 023.

`awslogs`descontinuado em favor do agente Amazon Logs unificado CloudWatch

O `awslogs`pacote está obsoleto AL2 e não está mais presente em 023. AL2 Ele é substituído pelo agente de CloudWatch registros unificado, disponível no `amazon-cloudwatch-agent` pacote. Para obter mais informações, consulte o Guia do usuário do Amazon CloudWatch Logs.

`bzr`sistema de controle de revisão

O sistema de controle de revisão GNU Bazaar(`bzr`) foi descontinuado AL2 e não está mais presente em AL2 023.

Os usuários do `bzr` são aconselhados a migrar seus repositórios para o. `git`

grupo v1

AL2023 passa para a hierarquia do Grupo de Controle Unificado (cgroup v2), enquanto AL2 usa cgroup v1. Como AL2 não é compatível com cgroup v2, essa migração precisa ser concluída como parte da mudança para AL2 023.

log4jhotpatch () `log4j-cve-2021-44228-hotpatch`

> **Note**
>
> O `log4j-cve-2021-44228-hotpatch` pacote foi descontinuado AL2 e removido em 023. AL2

Em resposta ao CVE-2021-44228, a Amazon Linux lançou uma versão empacotada em RPM do Hotpatch para Apache Log4j para e. AL1 AL2 No anúncio da adição do hotpatch ao Amazon Linux, observamos que “Instalar o hotpatch não substitui a atualização para uma versão log4j que atenua o CVE-2021-44228 ou o CVE-2021-45046.”.

O hotpatch foi uma mitigação para dar tempo de corrigir `log4j`. A primeira versão de disponibilidade geral do AL2 023 foi 15 meses após o CVE-2021-44228, portanto, o AL2 023 não vem com o hotpatch (ativado ou não).

Os clientes que executam suas próprias versões `log4j` no Amazon Linux são aconselhados a garantir que tenham atualizado para versões não afetadas pela CVE-2021-44228 ou CVE-2021-45046.

`lsb_release` e o pacote `system-lsb-core`

Historicamente, alguns softwares invocavam o `lsb_release` comando (fornecido AL2 pelo `system-lsb-core` pacote) para obter informações sobre a distribuição Linux na qual ele estava sendo executado. O Linux Standards Base (LSB) introduziu esse comando e as distribuições Linux o adotaram. As distribuições Linux evoluíram para usar o padrão mais simples de armazenar essas informações em `/etc/os-release` e outros arquivos relacionados.

O padrão `os-release` sai de `systemd`. Para obter mais informações, consulte a documentação do systemd os-release.

AL2023 não vem com o `lsb_release` comando e não inclui o `system-lsb-core` pacote. O software deve concluir a transição para o padrão `os-release` para manter a compatibilidade com o Amazon Linux e outras grandes distribuições Linux.

`mcrypt`

A `mcrypt` biblioteca e a PHP extensão associada foram descontinuadas em AL2 e não estão mais presentes em 023. AL2

O Upstream PHP descontinuou a `mcrypt` extensão na PHP versão 7.1, que foi lançada pela primeira vez em dezembro de 2016 e teve seu lançamento final em outubro de 2019.

A `mcrypt` biblioteca upstream foi lançada pela última vez em 2007 e não fez a migração do controle de `cvs` revisão SourceForge exigida para novos commits em 2017, com o commit mais recente (e apenas 3 anos antes) sendo de 2011, removendo a menção de que o projeto tinha um mantenedor.

Todos os usuários restantes do `mcrypt` são aconselhados a portar seu código paraOpenSSL, pois não `mcrypt` será adicionado ao AL2 023.

OpenJDK (7) `java-1.7.0-openjdk`

> **Note**
>
> AL2O 023 fornece várias versões do Amazon Corretto para Java suportar cargas de trabalho baseadas. Os pacotes do OpenJDK 7 estão obsoletos e não estão mais presentes em 023 AL2. AL2 O JDK mais antigo disponível em AL2 023 é fornecido pelo Corretto 8.

Para obter mais informações sobre Java no Amazon Linux, consulteJavaem AL2 023.

Python 2.7

> **Note**
>
> AL2023 removeu o Python 2.7, então todos os componentes do sistema operacional que exigem Python são escritos para funcionar com o Python 3. Para continuar usando uma versão do Python fornecida e compatível com o Amazon Linux, converta o código do Python 2 em Python 3.

Para obter mais informações sobre Python no Amazon Linux, consulte. Python em AL2 023

`rsyslog-openssl`substitui `rsyslog-gnutls`

O `rsyslog-gnutls` pacote está obsoleto em AL2 e não está mais presente em 023. AL2 O `rsyslog-openssl` pacote deve ser um substituto imediato para qualquer uso do `rsyslog-gnutls` pacote.

Serviço de informações de rede (NIS)/`yp`

O Network Information Service (NIS), originalmente chamado de Páginas Amarelas ou YP está obsoleto em AL2, e não está mais presente em 023. AL2 Isso inclui os seguintes pacotes: `ypbind`, `ypserv`, `yp-tools` e. Outros pacotes que se integram ao NIS tiveram essa funcionalidade removida em AL2 023.

Vários nomes de domínio na Amazon VPC `create-dhcp-options`

No Amazon Linux 2, era possível passar vários nomes de domínio no `domain-name` parâmetro para `create-dhcp-options`, o que resultaria em `/etc/resolv.conf` conter algo parecido`search foo.example.com bar.example.com`. O DHCP servidor Amazon VPC envia a lista de nomes de domínio fornecidos usando a DHCP opção 15, que suporta apenas um único nome de domínio (consulte a seção 3.17 da RFC 2132). Como AL2 023 usa `systemd-networkd` para configuração de rede, que segue aRFC, esse recurso acidental em não AL2 está presente em 023 AL2

A documentação AWS CLIe o Amazon VPC dizem o seguinte: “Alguns sistemas operacionais Linux aceitam vários nomes de domínio separados por espaços. No entanto, Windows outros sistemas operacionais Linux tratam o valor como um único domínio, o que resulta em um comportamento inesperado. Se seu conjunto de DHCP opções estiver associado a uma Amazon VPC que tenha instâncias executando sistemas operacionais que tratam o valor como um único domínio, especifique somente um nome de domínio. “

Nesses sistemas, como AL2 023, especificar dois domínios usando a DHCP opção 15 (que permite apenas um) e, como o caractere de espaço é inválido em nomes de domínio, isso resultará na codificação do caractere de espaço como`032`, resultando em conter. `/etc/resolv.conf` `search foo.exmple.com032bar.example.com`

Para oferecer suporte a vários nomes de domínio, um DHCP servidor deve usar a DHCP Opção 119 (consulte RFC 3397, seção 2). Consulte o Guia do usuário da Amazon VPC para saber quando isso é suportado pelo servidor Amazon VPC. DHCP

Sun RPC no **`glibc`**

A implementação de Sun RPC in `glibc` foi descontinuada AL2 e removida em 023. AL2 Recomenda-se que os clientes passem a usar a `libtirpc` biblioteca (disponível em AL2 e AL2 023) se a Sun RPC funcionalidade for necessária. A adoção `libtirpc` também permite que os aplicativos ofereçam suporteIPv6.

Essa mudança reflete a adoção mais ampla da comunidade da `glibc` remoção dessa funcionalidade pelo upstream, por exemplo, a remoção de Sun RPC interfaces do `glibc` Fedora e uma mudança semelhante no Gentoo.

Impressão digital da chave OpenSSH no registro **`audit`**

Posteriormente no ciclo de vida do AL2, um patch foi adicionado ao pacote OpenSSH para emitir a impressão digital da chave usada para autenticar. Essa funcionalidade não está presente no AL2 023.

`ld.gold`Vinculador

O `ld.gold` vinculador está disponível em AL2 e é removido em AL2 023. Os clientes que criam software que faça referência explícita ao gold vinculador devem migrar para o vinculador regular ()`ld.bfd`.

As notas de lançamento upstream do GNU Binutils para a versão 2.44 (lançada em fevereiro de 2025) documentam a remoção de`ld.gold`: “Em uma mudança em nossa prática anterior, nesta versão, o tarball binutils-2.44.tar não contém as fontes do vinculador gold. Isso ocorre porque o gold linker agora está obsoleto e, eventualmente, será removido, a menos que os voluntários se apresentem e se ofereçam para continuar o desenvolvimento e a manutenção.”

ping6

No AL2 023, o `ping` utilitário regular oferece suporte IPv6 nativo e o separado não `/bin/ping6` é mais necessário. Em AL2 023, `/usr/sbin/ping6` é um link simbólico para o `/usr/bin/ping` executável.

Essa mudança segue a adoção pela comunidade mais ampla de `iputils` versões mais recentes que fornecem essa funcionalidade, por exemplo, a IPv6 mudança de Ping no Fedora.

Obsoleto em 2023 AL2

Esta seção descreve a funcionalidade que existe em AL2 023 e provavelmente será removida em uma versão futura do Amazon Linux. Cada seção descreverá qual é a funcionalidade e quando se espera que ela seja removida do Amazon Linux.

> **Note**
>
> Esta seção será atualizada com o tempo, à medida que o ecossistema Linux evoluir e as futuras versões principais do Amazon Linux estiverem mais próximas do lançamento.

Tópicos

Suporte de tempo de execução x86 (i686) de 32 bits

AL2O 023 mantém a capacidade de executar binários x86 (i686) de 32 bits. É provável que a próxima versão principal do Amazon Linux não ofereça mais suporte à execução de binários de espaço de usuário de 32 bits.

`aspell`

Embora o AL2 023 seja fornecido com o `aspell` pacote, ele está obsoleto e será removido na próxima versão principal do Amazon Linux. Os clientes são aconselhados a migrar para substitutos modernos, como `hunspell` ou. `enchant2`

A descontinuação do `aspell` in AL2 023 segue uma mudança mais ampla da comunidade, por exemplo, a depreciação em. `aspell`Fedora

Berkeley DB (2) `libdb`

AL2O 023 vem com a versão 5.3.28 da biblioteca Berkeley DB (). `libdb` Esta é a última versão do Berkeley DB antes da mudança da licença para a licença GNU Affero GPLv3 (AGPL), da licença menos restritiva do Sleepycat.

Há poucos pacotes em AL2 023 que permanecem dependentes do Berkeley DB (`libdb`), e a biblioteca será removida na próxima versão principal do Amazon Linux.

> **Note**
>
> O gerenciador de `dnf` pacotes em AL2 023 mantém suporte somente para leitura para um banco de dados no formato Berkeley DB (BDB). `rpm` Esse suporte será removido na próxima versão principal do Amazon Linux.

A descontinuação de `libdb` segue a mudança mais ampla da comunidade, por exemplo, a depreciação em. `libdb`Fedora

`cron`

O `cronie` pacote foi instalado por padrão na AL2 AMI, fornecendo suporte para a `crontab` forma tradicional de programar tarefas periódicas. Em AL2 023, não `cronie` está incluído por padrão. Portanto, o suporte para não `crontab` é mais fornecido por padrão.

Em AL2 023, você pode instalar opcionalmente o `cronie` pacote para usar trabalhos clássicos`cron`. Recomendamos que você migre para temporizadores `systemd` devido à funcionalidade adicional fornecida pelo `systemd`.

É possível que uma versão futura do Amazon Linux, possivelmente a próxima versão principal, não inclua mais suporte para `cron` trabalhos clássicos e conclua a transição para `systemd` temporizadores. Recomendamos que você deixe de usar`cron`.

IMDSv1

Por padrão, AL2 023 AMIs são configurados para iniciar somente no modo IMDSv2 -only, desabilitando o uso de. IMDSv1 Ainda existe a opção de usar AL2 023 com IMDSv1 ativado. É provável que uma versão futura do Amazon Linux seja aplicada IMDSv2 somente.

Para obter mais informações sobre a configuração do IMDS para AMIs, consulte Configurar a AMI no Guia do EC2 usuário da Amazon.

`pcre` versão 1

O `pcre` pacote legado está obsoleto e será removido na próxima versão principal do Amazon Linux. O pacote `pcre2` é o sucessor. Embora as primeiras versões do AL2 023 tenham sido fornecidas com um número limitado de pacotes compilados`pcre`, esses pacotes serão migrados para `pcre2` o 023. AL2 A `pcre` biblioteca obsoleta permanecerá disponível em 2023. AL2

> **Note**
>
> A versão obsoleta do não `pcre` receberá atualizações de segurança durante toda a vida útil do 023. AL2 Para obter mais informações sobre o ciclo de vida do `pcre` suporte e a quantidade de tempo em que o pacote receberá atualizações de segurança, consulte as declarações de suporte do `pcre` pacote.

A descontinuação de `pcre` em favor de `pcre2` segue uma mudança mais ampla da comunidade nessa direção, por exemplo`pcre`, a depreciação em. Fedora

System V init (`sysvinit`)

Embora o AL2 023 mantenha a compatibilidade com versões anteriores dos scripts System V service (init), o `systemd` projeto upstream, como parte de sua versão v254, anunciou a descontinuação do suporte aos scripts de serviço System V e indicou que o suporte será removido em uma versão futura do. `systemd` Para obter mais informações, consulte systemd.

AL2O 023 manterá a compatibilidade com versões anteriores dos scripts System V service (init), mas os usuários são incentivados a migrar para o uso de arquivos `systemd` unitários nativos para estarem preparados para quando o suporte aos scripts System V service (init) for removido do Amazon Linux, provavelmente na próxima versão principal.

Pacotes EOL estão obsoletos

Cada pacote disponível em AL2 023 tem uma declaração de suporte associada que abrange informações específicas do Amazon Linux. Essas declarações abrangem o núcleo do sistema operacional e sua vida útil, bem como pacotes como the section called “PHP” ethe section called

“Python”, em que o AL2 023 vem com várias versões e cada uma é suportada pela duração do projeto de código aberto upstream.

Em AL2 023, você pode obter informações de suporte de pacotes usando o gerenciador de `dnf` pacotes. Para obter mais informações, consulte Obtendo informações de suporte do pacote.

Quando um pacote não é mais suportado antes do final da versão principal do Amazon Linux, deve-se presumir que esse pacote está obsoleto e não estará presente na próxima versão principal do Amazon Linux.

Para pacotes como the section called “PHP” ethe section called “Python”, em que cada versão principal do Amazon Linux foi fornecida com várias versões, cada uma com um ciclo de vida de suporte diferente, é provável que continuem presentes nas novas versões principais do Amazon Linux, embora com pouca ou nenhuma sobreposição das versões principais dos pacotes. É recomendável ter em mente os cronogramas de suporte do pacote Amazon Linux ao selecionar dependências.

Comparando AL2 e AL2 023

Os tópicos a seguir descrevem as principais diferenças entre AL2 e AL2 023.

Para obter mais informações sobre a funcionalidade obsoleta em AL1,, e AL2 AL2 023, consulte. Funcionalidade obsoleta em 2023 AL2

Tópicos

- Inicialização preferencial e segura por UEFI
- SSH alterações na configuração padrão do servidor
- AL2023 mudanças no kernel de AL2
- /tmp é agora tmpfs
- Alterações na AMI e na imagem do contêiner
- Comparando pacotes instalados no Amazon Linux 2 e no Amazon Linux 2023 AMIs
- Comparando pacotes instalados no Amazon Linux 2 e no Amazon Linux 2023 Minimal AMIs
- Compare de comparação de pacotes instalados em imagens de contêiner de base do Amazon Linux 2023 e do Amazon Linux 2023

Pacotes adicionados, atualizados e removidos

AL2023 contém milhares de pacotes de software disponíveis para uso. Para obter uma lista completa de todos os pacotes adicionados, atualizados ou removidos em AL2 023 em comparação com as versões anteriores do Amazon Linux, consulte Package changes in AL2 023.

Para solicitar que um pacote seja adicionado ou alterado em AL2 023, registre um problema no repositório amazon-linux-2023 em. GitHub

Suporte para cada versão

Para AL2 2023, oferecemos cinco anos de suporte.

Para obter mais informações, consulte Cadência de lançamento.

Alterações de nomenclatura e controle de versão

AL2O 023 suporta os mesmos mecanismos que AL2 suportam a identificação da plataforma. AL2O 023 também introduz novos arquivos para identificação da plataforma.

Para obter mais informações, consulte Nomeação e controle de versão.

Otimizações

AL2O 023 otimiza o tempo de inicialização para reduzir o tempo desde a inicialização da instância até a execução da carga de trabalho do cliente. Essas otimizações abrangem a configuração, as

`cloud-init` configurações e os recursos do kernel da EC2 instância Amazon que são incorporados aos pacotes no sistema operacional, como e. `kmod` `systemd`

Para obter mais informações sobre essas otimizações, consulte Otimizações operacionais e de desempenho.

Proveniente de vários upstreams

AL2O 023 é baseado em RPM e inclui componentes provenientes de várias versões do Fedora e de outras distribuições, como o CentOS 9 Stream. O kernel Amazon Linux é originado das versões de suporte de longo prazo (LTS) diretamente do kernel.org, escolhidas independentemente de outras distribuições.

Para obter mais informações, consulte Relacionamento com o Fedora.

Serviço do sistema de rede

O serviço `systemd-networkd` do sistema gerencia as interfaces de rede em AL2 023. Esta é uma mudança de AL2, que usa ISC dhclient ou no `dhclient`.

Para obter mais informações, consulte Serviço de redes.

Gerenciador de pacote

A ferramenta padrão de gerenciamento de pacotes de software em AL2 023 é DNF. DNF é o sucessor de YUM, a ferramenta de gerenciamento de pacotes em AL2.

Para obter mais informações, consulte Ferramenta de gerenciamento de pacotes.

Utilizar cloud-init

Em AL2 203, cloud-init gerencia o repositório de pacotes. Por padrão, em versões anteriores do Amazon Linux, cloud-init atualizações de segurança instaladas. Esse não é o padrão para AL2 023. Os novos recursos determinísticos de atualização para atualização `releasever` no lançamento descrevem a maneira AL2 023 de habilitar atualizações de pacotes no lançamento. Para obter mais informações, consulte Gerencie atualizações de pacotes e sistemas operacionais em AL2 023 e Atualizações determinísticas para estabilidade.

Com AL2 023, você pode usar cloud-init por SELinux. Para obter mais informações, consulteUse cloud-init para ativar o `enforcing` modo.

Cloud-init carrega o conteúdo da configuração com cloud-init de locais remotos usando HTTP(S). Nas versões anteriores, o Amazon Linux não alertava você quando os recursos remotos não estão disponíveis. Em AL2 023, recursos remotos indisponíveis criam um erro fatal e falham no cloud-init execução. Essa mudança no comportamento de AL2, fornece um comportamento padrão mais seguro de “falha fechada”.

Para obter mais informações, consulte Personalizado cloud-init e o cloud-init Documentação.

Suporte gráfico para desktop

AL2O 023 apresenta um ambiente de desktop gráfico baseado em GNOME a partir da versão 2023.7, substituindo o desktop MATE usado em. AL2 Esta versão oferece aos usuários uma experiência de desktop diferente, mantendo o desempenho otimizado para nuvem do AL2 023. O ambiente de trabalho GNOME oferece várias opções de personalização, recursos de integração do sistema e um design de interface de usuário distinto, fornecendo aos usuários uma alternativa ao ambiente de desktop MATE anterior. Consulte a AL2023 Desktop gráfico página para obter mais detalhes.

Compilador Triplet

AL2023 define o trigêmeo do compilador para GCC e `LLVM` para indicar que `amazon` é o fornecedor.

Assim, eles se AL2 `aarch64-redhat-linux-gcc` tornam `aarch64-amazon-linux-gcc` em AL2 023.

Isso deve ser completamente transparente para a maioria dos usuários e pode afetar apenas aqueles que estão criando compiladores no AL2 023.

Pacotes x86 (i686) de 32 bits

Como parte da versão 2014.09 AL1, foi anunciado que seria a última versão a produzir 32 bits. AMIs Portanto, a partir da versão 2015.03 do, o AL1 Amazon Linux não suportava mais a execução do sistema no modo de 32 bits. AL2 ofereceu suporte de tempo de execução limitado para binários de 32 bits em hosts x86-64 e não forneceu pacotes de desenvolvimento para permitir a criação de

novos binários de 32 bits. AL2023 não inclui mais nenhum pacote de espaço de usuário de 32 bits. Recomendamos que você conclua sua transição para o código de 64 bits.

Se você precisar executar binários de 32 bits no AL2 023, é possível usar o espaço do usuário de 32 bits de AL2 dentro de um AL2 contêiner executado sobre o 023. AL2

`lsb_release` e o pacote `system-lsb-core`

Historicamente, alguns softwares invocavam o `lsb_release` comando (fornecido AL2 pelo `system-lsb-core` pacote) para obter informações sobre a distribuição Linux na qual ele estava sendo executado. O Linux Standards Base (LSB) introduziu esse comando e as distribuições Linux o adotaram. As distribuições Linux evoluíram para usar o padrão mais simples de armazenar essas informações em `/etc/os-release` e outros arquivos relacionados.

O padrão `os-release` sai de `systemd`. Para obter mais informações, consulte a documentação do systemd os-release.

AL2023 não vem com o `lsb_release` comando e não inclui o `system-lsb-core` pacote. O software deve concluir a transição para o padrão `os-release` para manter a compatibilidade com o Amazon Linux e outras grandes distribuições Linux.

Extra Packages for Enterprise Linux (EPEL)

> **Warning**
>
> O AL2 `epel` Extra habilitou o terceiro EPEL7 repositório. A partir de 2024-06-30, o terceiro EPEL7 o repositório não está mais sendo mantido.
> Esse repositório de terceiros não terá atualizações futuras. Isso significa que não haverá correções de segurança para pacotes no repositório EPEL.
> Esta seção abordará as opções em AL2 0.2.3 para alguns pacotes encontrados em EPEL.

Extra Packages for Enterprise Linux (EPEL) é um projeto no Fedora comunidade com o objetivo de criar uma grande variedade de pacotes para sistemas operacionais Linux de nível corporativo. O projeto produziu principalmente RHEL and CentOS pacotes. AL2 apresenta um alto nível de compatibilidade com CentOS 7. Como resultado, muitos EPEL7 os pacotes funcionam em AL2.

Atualmente, não há um EPEL or EPEL-semelhante a um repositório para AL2 023. No entanto, há vários pacotes que estavam em EPEL7 que os clientes usaram e AL2 que estão disponíveis em AL2

023 ou têm alternativas em AL2 023. Esta seção abordará alguns desses pacotes e quais são as opções em AL2 023.

> **Warning**
>
> Adicione somente repositórios projetados para serem usados com AL2 023.
> Embora os repositórios projetados para outras distribuições possam funcionar atualmente, não há garantia de que continuarão funcionando com qualquer atualização de pacote no AL2 023 ou com o repositório não projetado para uso com o 023. AL2

Também existem pacotes que podem ser instalados a partir de EPEL no AL2 qual não será adicionado a AL2 023. Os motivos comuns para isso são problemas como o projeto upstream não ser mais mantido ou não ser corrigido CVEs. Esta seção também abordará alguns desses pacotes e quais alternativas existem.

Tópicos

- axel- Cliente HTTP/FTP
- brotlie libbrotli - compressão
- collectd- Daemon de coleta de estatísticas
- cpulimit- Limitador de uso da CPU
- exim- agente de transferência de correio
- fuse3- Sistema de arquivos no espaço do usuário (FUSE) v3
- ganglia- Sistema de monitoramento distribuído
- git-lfs- controle de versão de arquivos grandes com o Git
- haveged- uma fonte de entropia usando o HAVEGE algoritmo
- inotify-tools- ferramentas de linha de comando inotify
- iperf- Referência de desempenho TCP/UDP
- jemalloc- malloc implementação alternativa
- libbsd- Biblioteca de funções compatível com BSD
- libserf- Biblioteca de cliente HTTP
- libzstd- biblioteca de compressão zstd

- lighttpdservidor web
- lshell- uma concha restrita
- monit- monitor de processos, arquivos, diretórios e dispositivos
- nodejs
- perl-Config-General
- python2-lockfile- bloqueio de arquivos
- python2-rsa- Python RSA puro
- python2-simplejson- Rotinas JSON para Python 2
- rkhunter- Caçador de rootkits
- rssh- um shell restrito para uso com o OpenSSH
- sscg- gerador de certificados SSL autoassinado
- stress- Teste de estresse
- stress-ng- Teste de estresse
- tmpwatch- remove arquivos com base na hora do último acesso
- xmlstarlet- utilitários XML de linha de comando

`axel`- Cliente HTTP/FTP

O `axel` pacote estava em EPEL7, e nunca foi enviado como parte do Amazon Linux. As alternativas disponíveis em AL2 023 são `curl` e. `wget`

> **Warning**
>
> A `-S` opção de `axel` usar um não criptografado http conexão para descobrir espelhos para um arquivo.

É altamente recomendável migrar qualquer uso do `axel` over para um `curl` ou`wget`.

`brotli`e `libbrotli` - compressão

Os `brotli` `libbrotli` pacotes da mão estavam em EPEL7, enquanto apenas o `brotli` pacote estava disponível no AL2 núcleo.

Tanto os `libbrotli` pacotes `brotli` quanto os estão incluídos em AL2 023.

O `brotli` pacote pode ser instalado no AL2 023 com o seguinte comando:

```
[ec2-user ~]$ sudo dnf install brotli
```

O `libbrotli` pacote pode ser instalado no AL2 023 com o seguinte comando:

```
[ec2-user ~]$ sudo dnf install libbrotli
```

`collectd`- Daemon de coleta de estatísticas

O `collect` pacote estava em EPEL7, e também estava disponível no `collectd` e `collectd-python3` AL2 Extras.

O `collectd` pacote está incluído no AL2 023 e pode ser instalado executando o seguinte comando:

```
[ec2-user ~]$ sudo dnf install collectd
```

`cpulimit`- Limitador de uso da CPU

No Amazon Linux 2023, `systemd` fornece funcionalidade para limitar o uso da CPU de processos ou grupos de processos. Essa funcionalidade também é fácil de usar para qualquer `systemd` serviço.

Existem recursos poderosos de controle de recursos fornecidos `systemd` que podem ser usados para garantir que qualquer tarefa ou grupo de tarefas seja limitado nos recursos que pode consumir. Para obter mais informações, consulte a documentação upstream systemd.resource-control, junto com o. Limitando o uso de recursos do processo em AL2 023 usando systemd

`exim`- agente de transferência de correio

O `exim` pacote estava em EPEL7, e anteriormente disponível em AL1. O Amazon Linux 2023 fornece tanto o Mail Transfer Agents `postfix` quanto o `sendmail` Mail Transfer Agents (MTAs).

`fuse3`- Sistema de arquivos no espaço do usuário (FUSE) v3

O `fuse3` pacote (incluindo `fuse3-libs` `efuse3-devel`) estava em EPEL7. Esses pacotes fazem parte do AL2 023 e cada um pode ser instalado executando o seguinte comando relevante:

```
[ec2-user ~]$ sudo dnf install fuse3
```

```
[ec2-user ~]$ sudo dnf install fuse3-libs
```

```
[ec2-user ~]$ sudo dnf install fuse3-devel
```

`ganglia`- Sistema de monitoramento distribuído

O `ganglia` pacote estava em EPEL7, e anteriormente disponível em AL1. Não foi enviado com AL2.

O projeto upstream teve um período de inatividade em que algumas vagas não CVEs estavam sendo atendidas. Embora tenha havido atividade recente no projeto upstream, não está planejado adicionar `ganglia` ao AL2 023.

`git-lfs`- controle de versão de arquivos grandes com o Git

O `git-lfs` pacote estava em EPEL7. No Amazon Linux 2023, o `git-lfs` pacote está incluído no repositório principal. Em AL2 023, `git-lfs` pode ser instalado executando o seguinte comando:

```
[ec2-user ~]$ sudo dnf install git-lfs
```

`haveged`- uma fonte de entropia usando o HAVEGE algoritmo

O `haveged` pacote estava em EPEL7. O Amazon Linux 2023 vem pré-configurado com fontes de entropia, não exigindo o uso de. `haveged`

`inotify-tools`- ferramentas de linha de comando inotify

O `inotify-tools` pacote estava em EPEL7, e está incluído em AL2 023.

> **Note**
>
> No AL2 023, `systemd` oferece suporte à ativação baseada em caminho, que pode ser usada para agir em eventos, como quando um caminho existe ou muda.
> Muito do que `inotify-tools` é usado agora pode ser melhor realizado de maneira mais confiável usando a ativação de `systemd` caminhos. Para obter mais informações, consulte systemd.path.

O `inotify-tools` pacote está incluído no AL2 023 e pode ser instalado executando o seguinte comando:

```
[ec2-user ~]$ sudo dnf install inotify-tools
```

`iperf`- Referência de desempenho TCP/UDP

O pacote da `iperf` versão 2 estava em EPEL7, e também estava disponível no `testing` AL2 Extra. e também estava disponível em AL1

> **Note**
>
> O `iperf3` pacote também está disponível, fornecendo a versão 3 do`iperf`.

O `iperf` pacote está incluído no AL2 023 e pode ser instalado executando o seguinte comando:

```
[ec2-user ~]$ sudo dnf install iperf
```

`jemalloc`- `malloc` implementação alternativa

O `jemalloc` pacote estava em EPEL7, e estava disponível no `lamp-mariadb10.2-php7.2` e `mariadb10.5` AL2 Extras.

O `jemalloc` pacote está incluído no AL2 023 e pode ser instalado executando o seguinte comando:

```
[ec2-user ~]$ sudo dnf install jemalloc
```

`libbsd`- Biblioteca de funções compatível com BSD

O `libbsd` pacote estava em EPEL7, e também estava disponível no `testing` AL2 Extra.

O `libbsd` pacote está incluído no AL2 023 e pode ser instalado executando o seguinte comando:

```
[ec2-user ~]$ sudo dnf install libbsd
```

Os arquivos de desenvolvimento do `libbsd` podem ser instalados executando o comando a seguir.

```
[ec2-user ~]$ sudo dnf install libbsd-devel
```

`libserf`- Biblioteca de cliente HTTP

O `libserf` pacote estava em EPEL7. O `libserf` pacote é fornecido no Amazon Linux 2023. Ele pode ser instalado executando o seguinte comando:

```
[ec2-user ~]$ sudo dnf install libserf
```

`libzstd`- biblioteca de compressão zstd

O `libzstd` pacote estava no AL2 núcleo, assim como no EPEL7. O `libzstd` pacote também faz parte do AL2 023.

```
[ec2-user ~]$ sudo dnf install libzstd
```

`lighttpd`servidor web

O `lighttpd` pacote estava em EPEL7, e anteriormente disponível em AL1. O Amazon Linux 2023 fornece tanto o Apache `httpd` quanto os servidores `nginx` web.

`lshell`- uma concha restrita

O `lshell` pacote nunca foi enviado como parte do Amazon Linux. Estava disponível em EPEL6. O repositório de pacotes do Fedora explica por `lshell` que ele não foi empacotado EPEL7 ou Fedora 30. Também foi removido do Debian.

O `lshell` projeto upstream não está mais sendo mantido ativamente e contém os conhecidos Críticos nãocorrigidos CVEs: CVE-2016-6902e CVE-2016-6903.

A alternativa sugerida no bug do Debian também não `rssh`é mantida no upstream, com o autor citando problemas de segurança não solucionáveis como o motivo.

Por esses motivos, `lshell` a adição a AL2 023 não está planejada.

`monit`- monitor de processos, arquivos, diretórios e dispositivos

No Amazon Linux 2023, `systemd` fornece uma ampla variedade de funcionalidades para monitorar, iniciar, interromper e reiniciar serviços. Isso inclui reinicializações com limite de taxa, espera entre

tentativas de reinicialização e inicialização de outro serviço em caso de falha. Para obter mais informações, consulte a documentação do systemd.service.

No AL2 023, `systemd` também oferece suporte à ativação baseada em caminho, que pode ser usada para agir em eventos, como quando um caminho existe ou muda. Para obter mais informações, consulte systemd.path.

Há opções de configuração comuns para `systemd` unidades que permitem especificar dependências, condicionais e ações a serem tomadas em caso de sucesso ou falha. Para obter mais informações, consulte a documentação do systemd.unit.

Existem recursos poderosos de controle de recursos fornecidos `systemd` que podem ser usados para garantir que qualquer tarefa de monitoramento não use CPU ou memória excessivas. Para obter mais informações, consulte systemd.resource-control.

`nodejs`

O pacote da `nodejs` versão 16 estava em EPEL7, e agora `nodejs` está incluído em AL2 023. No momento em que este artigo foi escrito, as `nodejs` versões 18 e 20 estavam disponíveis em AL2 023. Você pode instalar `nodejs` 18 em AL2 023 com o seguinte comando:

```
[ec2-user ~]$ sudo dnf install nodejs
```

Você pode instalar `nodejs` 20 em AL2 023 com o seguinte comando:

```
[ec2-user ~]$ sudo dnf install nodejs20
```

`perl-Config-General`

O `perl-Config-General` pacote estava em EPEL7, e agora está incluído em AL2 023. Você pode instalar o `perl-Config-General` pacote em AL2 023 com o seguinte comando:

```
[ec2-user ~]$ sudo dnf install perl-Config-General
```

Os módulos Perl também podem ser instalados perguntando DNF para instalar o pacote que fornece um módulo Perl específico. Com esse método, você pode usar o nome mais familiar do módulo Perl em vez do nome do pacote do sistema operacional.

```
[ec2-user ~]$ sudo dnf install 'perl(Confg::General)'
```

`python2-lockfile`- bloqueio de arquivos

O `python2-lockfile` pacote estava em EPEL7, e AL2 incluiu um `python-lockfile` pacote. Em AL2 023O Python 2.7 foi substituído pelo Python 3, portanto, uma variante do Python 2 desse pacote não será adicionada ao AL2 023.

A versão Python 3 deste pacote está incluída na AL2 versão 023. Você pode instalar o `python3-lockfile` pacote no AL2 023 com um dos seguintes comandos:

```
[ec2-user ~]$ sudo dnf install python3-lockfile
```

Módulos Python também podem ser instalados perguntando DNF para instalar o pacote que fornece um módulo Python específico.

```
[ec2-user ~]$ sudo dnf install 'python3dist(lockfile)'
```

`python2-rsa`- Python RSA puro

O `python2-rsa` pacote estava em EPEL7, e AL2 incluiu um `python2-rsa` pacote. Em AL2 023O Python 2.7 foi substituído pelo Python 3, portanto, uma variante do Python 2 desse pacote não será adicionada ao AL2 023.

A versão Python 3 deste pacote está incluída na AL2 versão 023. Você pode instalar o `python3-rsa` pacote no AL2 023 com um dos seguintes comandos:

```
[ec2-user ~]$ sudo dnf install python3-rsa
```

Módulos Python também podem ser instalados perguntando DNF para instalar o pacote que fornece um módulo Python específico.

```
[ec2-user ~]$ sudo dnf install 'python3dist(rsa)'
```

`python2-simplejson`- Rotinas JSON para Python 2

O `python2-simplejson` pacote estava em EPEL7. Em AL2 023O Python 2.7 foi substituído pelo Python 3, portanto, uma variante do Python 2 desse pacote não será adicionada ao AL2 023.

A versão Python 3 deste pacote está incluída na AL2 versão 023. Você pode instalar o `python3-simplejson` pacote em AL2 023 com o seguinte comando:

```
[ec2-user ~]$ sudo dnf install python3-simplejson
```

Módulos Python também podem ser instalados perguntando DNF para instalar o pacote que fornece um módulo Python específico.

```
[ec2-user ~]$ sudo dnf install 'python3dist(simplejson)'
```

`rkhunter`- Caçador de rootkits

O `rkhunter` pacote está incluído em AL2 023 junto com`chkrootkit`.

```
[ec2-user ~]$ sudo dnf install rkhunter
```

```
[ec2-user ~]$ sudo dnf install chkrootkit
```

`rssh`- um shell restrito para uso com o OpenSSH

O `rssh` pacote estava em EPEL7. O `rssh`pacote upstream não é mantido, com o autor citando problemas de segurança não solucionáveis como o motivo.

Com o autor citando problemas de segurança não solucionáveis, a adição `rssh` a AL2 023 não está planejada.

`sscg`- gerador de certificados SSL autoassinado

O `sscg` pacote estava no AL2 núcleo, assim como no EPEL7. O `sscg` pacote também faz parte do AL2 023.

```
[ec2-user ~]$ sudo dnf install sscg
```

`stress`- Teste de estresse

O `stress` pacote estava em EPEL7, e também estava disponível em AL1

O `stress` pacote está incluído no AL2 023 e pode ser instalado executando o seguinte comando:

```
[ec2-user ~]$ sudo dnf install stress
```

`stress-ng`- Teste de estresse

O `stress-ng` pacote estava em EPEL7, e também estava disponível no `testing` AL2 Extra.

O `stress-ng` pacote está incluído no AL2 023 e pode ser instalado executando o seguinte comando:

```
[ec2-user ~]$ sudo dnf install stress-ng
```

`tmpwatch`- remove arquivos com base na hora do último acesso

No Amazon Linux 2023, essa funcionalidade é fornecida pelo `systemd-tmpfiles`.

`xmlstarlet`- utilitários XML de linha de comando

O `xmlstarlet` pacote estava em EPEL7, e não está disponível em AL2 023.

O pacote upstream não foi tocado em mais de 9 anos (a última atualização foi em agosto de 2014). Por mais quatro anos (desde pelo menos julho de 2010), uma solicitação por um novo mantenedor ficou sem resposta. É por esse motivo que não está planejado `xmlstarlet` adicionar AL2 023.

O Python 2.7 foi substituído pelo Python 3

AL2 fornece patches de suporte e segurança para o Python 2.7 até junho de 2025, como parte do nosso compromisso de suporte de longo prazo (LTS) para AL2 pacotes principais. Esse suporte vai além da declaração da comunidade Python upstream do Python 2.7 de janeiro de 2020 end-of-life.

AL2 usa o gerenciador de `yum` pacotes, que tem uma forte dependência do Python 2.7. Em AL2 023, o gerenciador de `dnf` pacotes migrou para o Python 3 e não precisa mais do Python 2.7. AL2023 foi completamente movido para o Python 3.

> **Note**
>
> AL2023 removeu o Python 2.7, então todos os componentes do sistema operacional que exigem Python são escritos para funcionar com o Python 3. Para continuar usando uma

versão do Python fornecida e compatível com o Amazon Linux, converta o código do Python 2 em Python 3.

Para obter mais informações sobre Python no Amazon Linux, consulte Python em AL2 023.

Atualizações de segurança

O Amazon Linux 2023 aprimora o endurecimento presente em. AL2 Para obter mais informações, consulte Segurança e compatibilidade no Amazon Linux 2023. Para obter mais informações sobre as alterações do fortalecimento do AL2 kernel, consulte. Alterações na configuração do kernel com foco na segurança

Tópicos

- SELinux
- OpenSSL 3
- IMDSv2
- Remoção de log4j hotpatch () log4j-cve-2021-44228-hotpatch

SELinux

Por padrão, Security Enhanced Linux (SELinux) para AL2 023 é `enabled` e definido como `permissive` modo. No modo `permissive`, as negações de permissão são registradas, mas não aplicadas.

SELinux é um recurso de segurança do kernel Amazon Linux, que estava `disabled` em AL2. SELinux é uma coleção de recursos e utilitários do kernel que fornece controle de acesso obrigatório (MAC) arquitetura nos principais subsistemas do kernel.

Para obter mais informações, consulte SELinux Modos de configuração para AL2 023.

Para obter mais informações sobre SELinux repositórios, ferramentas e políticas, consulte SELinux Notebook, Tipos de SELinux política e SELinux Projeto.

OpenSSL 3

AL2023 apresenta o Open Secure Sockets Layer version 3 (OpenSSL 3) kit de ferramentas de criptografia. AL2023 suportes TLS 1.3 and TLS 1.2 protocolos de rede.

Por padrão, AL2 vem com OpenSSL 1.0.2. Você pode criar aplicativos contra OpenSSL 1.1.1.

Para obter mais informações sobre OpenSSL, veja o OpenSSL guia de migração.

Para obter mais informações sobre a segurança, consulte Atualizações e recursos de segurança.

IMDSv2

Por padrão, todas as instâncias lançadas com a AMI AL2 023 exigem IMDSv2-only e seu limite de salto padrão será definido como 2 para permitir o suporte à carga de trabalho em contêineres. Isso é feito definindo o parâmetro `imds-support` como `v2.0`. Para obter mais informações, consulte Configurar a AMI no Guia EC2 do usuário da Amazon.

> **Note**
>
> O tempo de validade do token de sessão pode estar entre 1 segundo e 6 horas. Os endereços para direcionar o API pedidos de IMDSv2 as consultas são as seguintes:
>
> - IPv4: 169.254.169.254
> - IPv6: fd00:ec2:254

Você pode substituir manualmente essas configurações e ativar IMDSv1 usando as propriedades de lançamento da opção Instance Metadata. Você também pode usar controles do IAM para impor diferentes IMDS configurações. Para obter mais informações sobre como configurar e usar o Instance Metadata Service, consulte Use IMDSv2, Configure as opções de metadados da instância para novas instâncias e Modifique as opções de metadados da instância para instâncias existentes, no Guia EC2 do usuário da Amazon.

Remoção de log4j hotpatch () **`log4j-cve-2021-44228-hotpatch`**

> **Note**
>
> AL2023 não é enviado com o `log4j-cve-2021-44228-hotpatch` pacote.

Em resposta ao CVE-2021-44228, a Amazon Linux lançou uma versão empacotada em RPM do Hotpatch para Apache Log4j para e. AL1 AL2 No anúncio da adição do hotpatch ao Amazon Linux,

observamos que “Instalar o hotpatch não substitui a atualização para uma versão log4j que atenua o CVE-2021-44228 ou o CVE-2021-45046”.

O hotpatch foi uma mitigação para dar tempo de corrigir `log4j`. A primeira versão de Disponibilidade Geral (GA) do AL2 023 foi 15 meses após o CVE-2021-44228, portanto, o AL2 023 não vem com o hotpatch (ativado ou não).

Os usuários que executam suas próprias `log4j` versões no Amazon Linux devem garantir que tenham atualizado para versões não afetadas pela CVE-2021-44228 ou CVE-2021-45046.

AL2O 023 fornece orientações Atualizando AL2 023 para que você possa se manter atualizado com os patches de segurança. Os avisos de segurança são publicados no Amazon Linux Security Center.

Atualizações determinísticas para estabilidade

Com o recurso de atualizações determinísticas por meio de repositórios versionados, cada 023 AL2 AMI, por padrão, está bloqueada para uma versão específica do repositório. Você pode usar atualizações determinísticas para obter maior consistência entre as versões e atualizações do pacote. Cada versão, principal ou secundária, inclui uma versão específica do repositório.

Novo no AL2 023, a atualização determinística por padrão está ativada. Essa é uma melhoria em relação ao método manual e incremental de bloqueio usado em AL2 e em outras versões anteriores.

Para obter mais informações, consulte Atualizações determinísticas por meio de repositórios versionados em 023 AL2.

gp3como tipo de volume padrão do Amazon EBS

A AMI AL2 023 e AL2 ambas usam o XFS sistema de arquivos no sistema de arquivos raiz. Para AL2 023, `mkfs` as opções do sistema de arquivos do dispositivo raiz são ainda mais otimizadas para a Amazon EC2. AL2O 023 também oferece suporte a vários outros sistemas de arquivos que você pode usar em outros volumes para atender aos seus requisitos específicos.

AL2023 AMIs usa `gp3` volumes Amazon EBS por padrão, enquanto usa `gp2` volumes AL2 AMIs Amazon EBS por padrão. É possível alterar o tipo de volume quando lançar uma instância.

Para obter mais informações sobre os tipos de volume Amazon EBS, consulte Volumes de propósito do Amazon EBS.

Para obter mais informações sobre o lançamento de uma EC2 instância da Amazon, consulte Iniciar uma instância no Guia EC2 do usuário da Amazon.

Hierarquia unificada do Grupo de Controle (cgroup v2)

Um grupo de controle (cgroup) é um recurso do kernel Linux para organizar processos hierarquicamente e distribuir recursos do sistema entre eles. Os grupos de controle são usados extensivamente para implementar um tempo de execução de contêiner e por `systemd`.

AL2 apoios cgroupv1, e AL2 023 suportes cgroupv2. Isso é notável ao executar cargas de trabalho em contêineres, como quando. Usando o Amazon ECS baseado em AL2 023 AMIs para hospedar cargas de trabalho em contêineres

Embora o AL2 023 ainda inclua código que pode fazer o sistema funcionar usando cgroupv1, essa não é uma configuração recomendada ou suportada e será completamente removida em uma futura versão principal do Amazon Linux.

Há uma extensa documentação sobre as interfaces de baixo nível de kernel do Linux, bem como a documentação de delegação de systemd cgroup.

Um caso de uso comum fora dos contêineres é criar `systemd` unidades que tenham limites nos recursos do sistema que elas podem usar. Para obter mais informações, consulte systemd.resource-control.

`systemd`temporizadores substituem **`cron`**

O `cronie` pacote foi instalado por padrão na AL2 AMI, fornecendo suporte para a `crontab` forma tradicional de programar tarefas periódicas. Em AL2 023, não `cronie` está incluído por padrão. Portanto, o suporte para não `crontab` é mais fornecido por padrão.

Opcionalmente, você pode instalar o pacote `cronie` para usar trabalhos tarefas clássicas `cron`. Recomendamos que você migre para temporizadores `systemd` devido à funcionalidade adicional fornecida pelo `systemd`.

Conjunto de ferramentas aprimorado: gcc, binutils e glibc

AL2O 023 inclui muitos dos mesmos pacotes principais do. AL2

Atualizamos os três pacotes principais do conjunto de ferramentas a seguir para AL2 023.

Nome do pacote	AL2	AL2023
glibc	2.26	2.34
gcc	7.3	11.3
binutils	2.29	2,39

Para obter mais informações, consulte Pacotes principais do conjunto de ferramentas glibc, gcc, binutils.

Para obter mais informações sobre essas otimizações, consulte Otimizações operacionais e de desempenho.

`systemd`diário substitui `rsyslog`

Em AL2 023, o pacote do sistema de registro foi alterado de AL2. AL2O 023 não é instalado `rsyslog` por padrão, portanto, os arquivos de log baseados em texto, como os `/var/log/messages` que estavam disponíveis em, AL2 não estão disponíveis por padrão. A configuração padrão para AL2 023 é`systemd-journal`, que pode ser examinada usando`journalctl`. Embora `rsyslog` seja um pacote opcional no AL2 023, recomendamos a nova `journalctl` interface `systemd` baseada e os pacotes relacionados. Para obter mais informações, consulte a página do manual `journalctl`.

A ferramenta systmed journal equivalente a alguns comumente usados syslog os comandos são abordados na tabela a seguir.

AL2 syslog command	AL2023 systemd journal equivalente
`[ec2-user ~]$ cat /var/log/messages`	`[ec2-user ~]$ journalctl`
`[ec2-user ~]$ tail -f /var/log/messages`	`[ec2-user ~]$ journalctl -f`
`[ec2-user ~]$ grep foo /var/log/messages`	`[ec2-user ~]$ journalctl | grep foo`

Dependências de pacotes minimizadas

O Amazon Linux 2023 minimiza o gráfico de dependência de muitos pacotes para fornecer um espaço menor para os aplicativos. As mudanças notáveis AL2 incluem os `gnupg-minimal` pacotes `curl-minimal` e, que reduzem significativamente o número de pacotes necessários, mantendo a funcionalidade comumente usada.

Tópicos

Alterações de pacotes para **`curl`** e **`libcurl`**

AL2023 separa os protocolos e funcionalidades comuns dos `libcurl` pacotes `curl` e em e. `curl-minimal` `libcurl-minimal` Isso reduz o espaço ocupado por disco, memória e dependência para a maioria dos usuários e é o pacote padrão para AL2 023 AMIs e contêineres.

Se a funcionalidade completa do `curl` for necessária, por exemplo, para suporte de `gopher://`, execute os seguintes comandos para instalar os `curl-full` e `libcurl-full`.

```
$ dnf swap libcurl-minimal libcurl-full
```

```
$ dnf swap curl-minimal curl-full
```

Guarda de Privacidade GNU (GNUPG)

AL2023 separa a funcionalidade mínima e completa do `gnupg2` pacote em `gnupg2-minimal` pacotes. `gnupg2-full` Por padrão, apenas o pacote `gnupg2-minimal` está instalado. Isso fornece a funcionalidade mínima necessária para verificar as assinaturas digitais nos pacotes `rpm`.

Para obter mais funcionalidades de `gnupg2`, como a capacidade de baixar chaves de um servidor de chaves, verifique se o pacote `gnupg2-full` está instalado. Execute o seguinte comando para trocar `gnupg2-minimal` por `gnupg2-full`.

```
$ dnf swap gnupg2-minimal gnupg2-full
```

Amazon Corretto como JVM padrão

AL2O 023 é fornecido com o Amazon Corretto como o Java Development Kit (JDK) padrão (e único). Todos Java pacotes baseados em AL2 023 são todos construídos com Amazon Corretto 17.

Se você estiver migrando de AL2, você pode fazer a transição sem problemas da `OpenJDK` versão equivalente para AL2 Amazon Corretto.

AWS CLI v2

AL2O 023 vem com a AWS CLI versão 2, enquanto que AL2 vem com a versão 1 do AWS CLI.

Inicialização preferencial e segura por UEFI

Por padrão, todas as instâncias iniciadas com a AMI AL2 023 em tipos de instância compatíveis com o firmware UEFI serão iniciadas no modo UEFI. Isso é feito definindo o parâmetro AMI do modo de inicialização como `uefi-preferred`. Para obter mais informações, consulte Modos de inicialização no Guia EC2 do usuário da Amazon.

Nos tipos de EC2 instância da Amazon que oferecem suporte ao UEFI Secure Boot, é possível habilitar o Secure Boot no Amazon Linux 2023. Para obter mais informações, consulte Inicialização segura UEFI em 023 AL2.

SSH alterações na configuração padrão do servidor

Para a AMI AL2 023, alteramos os tipos de chaves de `sshd` host que geramos com a versão. Também eliminamos alguns tipos de chaves legadas para evitar gerá-las no momento do lançamento. Os clientes devem oferecer suporte aos protocolos `rsa-sha2-256` e `rsa-sha2-512` ou `ssh-ed25519` com o uso de uma chave `ed25519`. Por padrão, as assinaturas `ssh-rsa` estão desabilitadas.

Além disso, AL2 023 configurações no `sshd_config` arquivo padrão contêm`UseDNS=no`. Essa nova configuração significa que DNS é menos provável que deficiências bloqueiem sua capacidade de estabelecer `ssh` sessões com suas instâncias. A desvantagem é que as entradas de linha `from=`*`hostname.domain`*`,`*`hostname.domain`* em seus arquivos `authorized_keys` não serão resolvidas. Como `sshd` não há mais tentativas de resolver os nomes DNS, cada *`hostname.domain`* valor separado por vírgula deve ser traduzido para um correspondente IP address.

Para obter mais informações, consulte Configuração do servidor SSH padrão.

AL2023 mudanças no kernel de AL2

AL2O 023 traz o kernel 6.1, bem como muitas mudanças de configuração para otimizar ainda mais o Amazon Linux para a nuvem. Para a maioria dos usuários, essas alterações devem ser completamente transparentes.

IPv4 TTL

O TTL para IPv4 é configurado via`sysctl`, com os valores padrão presentes em`/etc/sysctl.d/00-defaults.conf`. Esse valor pode ser personalizado por meio dos `sysctl` métodos usuais. Para obter mais informações, consulte a `sysctl` `man` página.

AL2 defina o `net.ipv4.ip_default_ttl` valor como 255, enquanto AL2 023 o define como 127. Isso alinha os padrões do Amazon Linux com outras grandes distribuições Linux. Não é recomendável alterar esse padrão sem uma necessidade demonstrada.

Alterações na configuração do kernel com foco na segurança

Opção do **CONFIG**	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6 1/ aarch64	AL2023/6 1/ x86_64	AL2023/6 12/ aarch64	AL2023/6. 12/ x86_64
CONFIG_I G_ON_DA _CORRUP ON	n	y	n	y	y	y	y	y
CONFIG_I FAULT_MI P_MIN_AI R	4096	4096	4096	4096	65536	65536	65536	65536
CONFIG_I VMEM	n	y	n	y	n	n	n	n
CONFIG_I VPORT	n	y	n	y	n	n	n	n

Opção do **CONFIG**	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/ aarch64	AL2023/6 1/ x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64
CONFIG_I RTIFY_SC RCE	n	y	n	y	y	y	y	y
CONFIG_I RDENED_U ERCOPY_I LLBACK	N/D	N/D	y	y	N/D	N/D	N/D	N/D
CONFIG_I IT_ON_AL OC_DEFAU T_ON	N/D	N/D	n	n	n	n	n	n
CONFIG_I IT_ON_FI E_DEFAUL _ON	N/D	N/D	n	n	n	n	n	n
CONFIG_I MMU_DEFA LT_DMA_S RICT	N/D	N/D	N/D	N/D	n	n	n	n
CONFIG_I ISC_AUTO OAD	y	y	y	y	n	n	n	n
CONFIG_S HED_CORI	N/D	N/D	N/D	N/D	N/D	y	N/D	y

Opção do **CONFIG**	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/ aarch64	AL2023/6 1/ x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64
CONFIG_S HED_STA(_END_CHI K_	n	y	n	y	y	y	y	y
CONFIG_S CURITY_I ESG_RES ICT_	n	n	n	n	y	y	y	y
CONFIG_S CURITY_S LINUX_DI ABLE_	y	y	y	y	n	n	N/D	N/D
CONFIG_S UFFLE_PA E_ALLOCA OR_	N/D	N/D	y	y	y	y	y	y
CONFIG_S AB_FREEI ST_HARDI ED_	n	y	y	y	y	y	y	y
CONFIG_S AB_FREEI ST_RANDO	n	n	y	y	y	y	y	y

x86-64 Alterações específicas na configuração do kernel focadas na segurança

Opção do **CONFIG**	AL2/4.14/x86_64	AL2/5.10/x86_64	AL2023/6,1/ x86_64	AL2023/6.12/ x86_64
CONFIG_AMD_IOMMU	y	y	y	y
CONFIG_AMD_IOMMU_V2	m	m	y	N/D
CONFIG_RANDOMIZE_MEMORY	N/D	y	y	y

aarch64 (ARM/Graviton) Alterações específicas na configuração do kernel focadas na segurança

Opção do **CONFIG**	AL2/4.14/ aarch64	AL2/5.10/ aarch64	AL2023/6.1/ aarch64	AL2023/6.12/ aarch64
CONFIG_ARM64_PTR_AUTH	N/D	y	y	y
CONFIG_ARM64_PTR_AUTH_KERNEL	N/D	N/D	y	y
CONFIG_ARM64_SW_TTBR0_PAN	y	y	y	y

/dev/mem, **/dev/kmem** e **/dev/port**

O Amazon Linux 2023 desativa`/dev/mem`, `CONFIG_DEVMEM` e `/dev/port` (e`CONFIG_DEVPORT`) completamente, com base nas restrições já existentes. AL2

O `/dev/kmem` código foi completamente removido do Linux no kernel 5.13 e, embora tenha sido desativado AL2, agora não é aplicável ao AL2 023.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

`FORTIFY_SOURCE`

AL2O 023 é ativado `CONFIG_FORTIFY_SOURCE` em todas as arquiteturas suportadas. Esse recurso é um recurso de fortalecimento da segurança. Onde o compilador pode determinar e validar os tamanhos do buffer, esse recurso pode detectar estouros de buffer em funções comuns de string e memória.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Carregamento automático do Line Discipline () `CONFIG_LDISC_AUTOLOAD`

O kernel AL2 023 não carregará automaticamente disciplinas de linha, como por exemplo, por software usando o. `TIOCSETDioctl`, a menos que a solicitação venha de um processo com as permissões. `CAP_SYS_MODULE`

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

`dmesg`acesso para usuários sem privilégios () `CONFIG_SECURITY_DMESG_RESTRICT`

Por padrão, o AL2 023 não permite que usuários sem privilégios acessem o. `dmesg`

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

SELinux `selinuxfs`desabilitar

AL2023 desativa a opção obsoleta do `CONFIG_SECURITY_SELINUX_DISABLE` kernel, que habilitou um método de tempo de execução de desativação antes SELinux do carregamento da política.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Outras alterações na configuração do kernel

Opção do **CONFIG**	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/ aarch64	AL2023/6 1/ x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64
CONFIG_I	100	250	100	250	100	100	100	100
CONFIG_I _CPUS_	4096	8192	4096	8192	4096	8192	4096	8192
CONFIG_I NIC_ON_(PS_	y	n	y	n	y	y	y	y
CONFIG_I NIC_ON_(PS_VALUI	1	0	1	0	1	1	1	1
CONFIG_I P_	m	m	m	m	n	n	n	n
CONFIG_S IP_	m	m	m	m	n	n	n	n
CONFIG_X N_PV_	N/D	y	N/D	n	N/D	n	N/D	n

CONFIG_HZ

AL2023 define `CONFIG_HZ` como 100 em ambas `x86-64` as `aarch64` plataformas.

CONFIG_NR_CPUS

AL2023 é definido `CONFIG_NR_CPUS` para um número mais próximo do número máximo de núcleos de CPU encontrados na Amazon EC2.

Pânico no OOPS

O kernel AL2 023 entrará em pânico quando entrar em loop. Esse recurso é equivalente à inicialização com `oops=panic` na linha de comando do kernel.

Um kernel oops é onde o kernel detectou um erro interno que pode afetar a confiabilidade adicional do sistema.

Suporte para PPP e SLIP

AL2023 não suporta os protocolos PPP ou SLIP.

Suporte para convidados do Xen PV

AL2O 023 não suporta a execução como convidado do Xen PV.

Suporte ao sistema de arquivos do kernel

Houve várias mudanças nos sistemas de arquivos nos quais o kernel AL2 suportará a montagem, junto com mudanças nos esquemas de particionamento que o kernel analisará.

Opção do **CONFIG**	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/ aarch64	AL2023/6 1/ x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64
CONFIG_A S_FS	n	m	n	m	n	n	n	n
CONFIG_A _RXRPC	n	m	n	m	n	n	n	n
CONFIG_B D_DISKLA EL	y	y	y	y	n	n	n	n
CONFIG_C AMFS	m	m	m	m	n	n	n	n

Opção do **CONFIG**	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/ aarch64	AL2023/6 1/ x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64
CONFIG_(AMFS_BL(KDEV	N/D	N/D	y	n	N/D	N/D	N/D	N/D
CONFIG_I _CLONE	N/D	N/D	n	n	n	n	n	n
CONFIG_I _ERA	m	n	m	n	n	n	n	n
CONFIG_I _INTEGR Y	n	m	n	m	m	m	m	m
CONFIG_I _LOG_WR ES	n	n	m	m	m	m	m	m
CONFIG_I _SWITCH	m	n	m	n	n	n	n	n
CONFIG_I _VERITY	m	n	m	n	n	n	n	n
CONFIG_I RYPT_FS	n	m	n	m	n	n	n	n
CONFIG_I FAT_FS	N/D	N/D	m	m	m	m	m	m
CONFIG_I T2_FS	n	m	n	m	n	n	n	n
CONFIG_I T3_FS	n	m	n	m	n	n	n	n

Opção do **CONFIG**	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/ aarch64	AL2023/6 1/ x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64
CONFIG_(S2_FS	m	m	m	m	n	n	n	n
CONFIG_I SPLUS_F	n	m	n	m	n	n	n	n
CONFIG_I S_FS	n	m	n	m	n	n	n	n
CONFIG_ S_FS	n	n	n	n	n	n	n	n
CONFIG_I M_PARTI ON	n	y	n	y	n	n	n	n
CONFIG_I C_PARTI ON	n	y	n	y	n	n	n	n
CONFIG_I S_V2	n	m	n	m	n	n	n	n
CONFIG_I FS_FS	n	m	n	n	n	n	n	n
CONFIG_I MFS_FS	n	m	n	m	n	n	n	n
CONFIG_S LARIS_X8 _PARTIT N	n	y	n	y	n	n	n	n

Opção do **CONFIG**	AL2/4.14/ aarch64	AL2/4.14/ x86_64	AL2/5.10/ aarch64	AL2/5.10/ x86_64	AL2023/6. 1/ aarch64	AL2023/6 1/ x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64
CONFIG_S UASHFS_Z TD	n	y	n	y	y	y	y	y
CONFIG_S N_PARTI ON	n	y	n	y	n	n	n	n

Suporte ao sistema de arquivos Andrew (AFS)

O kernel não é mais construído com suporte para o sistema de `afs` arquivos. AL2 não foi fornecido com suporte de espaço de usuário para`afs`.

suporte cramfs

O kernel não é mais construído com suporte para o sistema de arquivos `cramfs`. O sucessor em AL2 023 é o sistema de `squashfs` arquivos.

Suporte a etiquetas de disco BSD

O kernel não é mais construído com suporte para rótulos de disco BSD. Se for necessário ler volumes com rótulos de disco BSD, vários BSDs podem ser iniciados.

Alterações no Device Mapper

Houve várias alterações nos alvos do Device Mapper configurados no kernel AL2 023.

eCryptFs apoio

O sistema de arquivos `ecryptfs` foi descontinuado no Amazon Linux. Os componentes do espaço de usuário do `ecryptfs` were present in AL1, removed in AL2 e AL2 023 não constroem mais o kernel com suporte. `ecryptfs`

exFAT

Support para o sistema de `exFAT` arquivos foi adicionado ao kernel 5.10 em. AL2 Ele não estava presente no AL2 lançamento com um kernel 4.14. AL2O 023 continua oferecendo suporte ao sistema de `exFAT` arquivos.

Os sistemas de arquivos ext2, ext3 e ext4

AL2O 023 vem com a `CONFIG_EXT4_USE_FOR_EXT2` opção, o que significa que o código do sistema de `ext4` arquivos será usado para ler sistemas de `ext2` arquivos legados.

CONFIG_ _FS GFS2

O kernel não é mais construído com CONFIG_ _FSGFS2.

Suporte estendido ao sistema de arquivos Apple HFS (HFS+)

Em AL2, somente os `x86-64` kernels foram construídos com o suporte do sistema de `hfsplus` arquivos. O kernel AL2 5.15 não inclui `hfsplus` suporte em nenhuma arquitetura. Em AL2 2013, concluímos a descontinuação do suporte `hfsplus` no Amazon Linux.

Suporte a sistemas de arquivos HFS

Em AL2, somente os `x86-64` kernels foram construídos com o suporte do sistema de `hfs` arquivos. O kernel AL2 5.15 não inclui `hfs` suporte em nenhuma arquitetura. Em AL2 2013, concluímos a descontinuação do suporte `hfs` no Amazon Linux.

Suporte a sistemas de arquivos JFS

Os AL2 `x86-64` kernels mais antigos foram construídos com suporte ao sistema `jfs` de arquivos. O kernel AL2 5.15 não inclui `jfs` suporte em nenhuma arquitetura. Nenhum deles AL1 ou AL2 fornecido com o espaço de usuário do JFS. Em AL2 2013, concluímos a descontinuação do suporte `jfs` no Amazon Linux.

O kernel Linux upstream está considerando a remoção do. `JFS` Portanto, se você tiver dados em um sistema de `JFS` arquivos, deverá migrá-los para outro sistema de arquivos. Em 2024, `JFS` foi removido de todos os kernels Linux atuais da Amazon.

WindowsSuporte ao Gerenciador de Disco Lógico (Disco Dinâmico**CONFIG_LDM_PARTITION**) ()

AL2023 não suporta Windows 2000 mais discos Windows Vista dinâmicos com partições de MS-DOS estilo. Windows XP Esse código nunca suportou os discos dinâmicos baseados em GPT mais recentes introduzidos com. Windows Vista

Suporte ao mapa de partições do Macintosh

AL2O 023 não suporta mais o mapa de partições clássico do Macintosh. As versões modernas do macOS criarão tabelas de partições GPT modernas por padrão sobre esse tipo mais antigo.

NFSv2 apoio

AL2023 não suporta mais NFSv2, mas continua suportando NFSv3 NFSv4, NFSv4 .1 e NFSv4 .2. Recomendamos que você migre para NFSv3 ou para uma versão mais recente.

NTFS (**CONFIG_NTFS_FS**)

O `ntfs3` código foi substituído `ntfs` para acessar sistemas de arquivos NTFS no Amazon Linux a partir do kernel 5.10 em. AL2 AL2O 023 não inclui mais o `ntfs` código e depende exclusivamente do `ntfs3` código para acessar sistemas de arquivos NTFS.

romfs file system

O sistema de `squashfs` arquivos é o sucessor do sistema de `romfs` arquivos no Amazon Linux, e o kernel AL2 023 não é mais construído com suporte para o. `romfs`

Formato de partição de disco rígido Solaris x86

AL2O 023 não oferece mais suporte ao formato de partição de disco rígido x86 do Solaris.

squashfszstd compressão

AL2023 adiciona suporte para sistemas de `squashfs` arquivos zstd compactados em todas as arquiteturas suportadas.

Suporte para tabela de partição Sun

AL2023 não inclui mais suporte para o formato de tabela de partições Sun (`CONFIG_SUN_PARTITION`).

`/tmp` é agora `tmpfs`

O Amazon Linux 2023 introduz mudanças na forma como se `/tmp` comporta em comparação com o Amazon Linux 2. A configuração padrão para AL2 era que ambos `/tmp` `/var/tmp` estivessem no sistema de arquivos raiz. O Amazon Linux 2023 usa como padrão o uso de `tmpfs` for `/tmp` com um limite de 50% de RAM e um máximo de um milhão inodes. Essas mudanças alinham o Amazon Linux com o comportamento de outras distribuições Linux.

Para obter detalhes completos sobre o layout do sistema de arquivos de AL2 023, consulte `/tmp` e `/var/tmp` na Layout do sistema de arquivos seção.

Alterações na AMI e na imagem do contêiner

Houve algumas mudanças nos pacotes incluídos AMIs e nos contêineres.

O Amazon Linux 2023 apresenta umthe section called “AL2023 Imagem mínima do contêiner”, e suporte para construçãothe section called “Criando imagens básicas de contêineres 023 AL2”. Para obter mais informações, consulte Usando AL2 023 em contêineres.

Comparando pacotes instalados no Amazon Linux 2 e no Amazon Linux 2023 AMIs

Uma comparação do RPMs presente no padrão AMIs Amazon Linux 2 e AL2 023.

Pacote	AL2 AMI	AL2203 AMI
`acl`	2.2.51	2.3.1
`acpid`	2.0.19	2.0.32
`alternatives`		1.15
`amazon-chrony-config`		4.3
`amazon-ec2-net-utils`		2.5.1
`amazon-linux-extras`	2.0.3	

Pacote	AL2 AMI	AL2203 AMI
amazon-linux-extras-yum-plugin	2.0.3	
amazon-linux-repo-s3		2023.6.20241031
amazon-linux-sb-keys		2023.1
amazon-rpm-config		228
amazon-ssm-agent	3.3.987.0	3.3.987.0
amd-ucode-firmware	2020421 (noarca)	20210208 (noarca)
at	3.1.13	3.1.23
attr	2.4.46	2.5.1
audit	2.8.1	3.0.6
audit-libs	2.8.1	3.0.6
authconfig	6.2.8	
aws-cfn-bootstrap	2,0	2.0
awscli	1.18.147	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
bash-completion	2.1	2.11
bc	1.06.95	1.07.1
bind-export-libs	9.11.4	
bind-libs	9.11.4	9.18.28

Pacote	AL2 AMI	AL2203 AMI
bind-libs-lite	9.11.4	
bind-license	9.11.4	9.18.28
bind-utils	9.11.4	9.18.28
binutils	2.29.1	2,39
blktrace	1.0.5	
boost-date-time	1.53.0 (x86_64)	
boost-filesystem		1.75.0
boost-system	1.53.0 (x86_64)	1.75.0
boost-thread	1.53.0 (x86_64)	1.75.0
bridge-utils	1.5	
bzip2	1.0.6	1.0.8
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.68	2023.2.68
c-ares		1.19.1
checkpolicy		3.4
chkconfig	1.7.4	1.15
chrony	4.2	4.3
cloud-init	19.3	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart	0,31	0,31

Pacote	AL2 AMI	AL2203 AMI
coreutils	8.22	8,32
coreutils-common		8,32
cpio	2.12	2.13
cracklib	2.9.0	2.9.6
cracklib-dicts	2.9.0	2.9.6
cronie	1.4.11	
cronie-anacron	1.4.11	
crontabs	1.11	1.11
crypto-policies		2020428
crypto-policies-scripts		2020428
cryptsetup	1.7.4	2.6.1
cryptsetup-libs	1.7.4	2.6.1
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	2.1.27
cyrus-sasl-plain	2.1.26	2.1.27
dbus	1.10.24	1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.10.24	1.12.28

Pacote	AL2 AMI	AL2203 AMI
device-mapper	1.02.170	1.02.185
device-mapper-event	1.02.170	
device-mapper-event-libs	1.02.170	
device-mapper-libs	1.02.170	1.02.185
device-mapper-persistent-data	0.7.3	
dhclient	4.2.5	
dhcp-common	4.2.5	
dhcp-libs	4.2.5	
diffutils	3.3	3.8
dmidecode	3.2	
dmraid	1.0.0.rc16	
dmraid-events	1.0.0.rc16	
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dnf-utils		4.3.0

Pacote	AL2 AMI	AL2203 AMI
dosfstools	3.0.20	4.2
dracut	033	055
dracut-config-ec2	2,0	3.0
dracut-config-generic	033	055
dwz		0,14
dyninst	9.3.1 (x86_64)	10.2.1
e2fsprogs	1.42.9	1.46,5
e2fsprogs-libs	1.42.9	1.46,5
ec2-hibinit-agent	1.0.8	1.0.8
ec2-instance-connect	1.1	1.1
ec2-instance-connect-selinux	1.1	1.1
ec2-net-utils	1.7.3	
ec2-utils	1.2	2.2.0
ed	1.9	1.14.2
efibootmgr	15 (março 64)	
efi-filesystem		5
efi-srpm-macros		5
efivar		38
efivar-libs	31 (março 64)	38

Pacote	AL2 AMI	AL2203 AMI
elfutils-debuginfod-client		0.188
elfutils-default-yama-scope	0,176	0.188
elfutils-libelf	0,176	0.188
elfutils-libs	0,176	0.188
ethtool	4.8	5.15
expat	2.1.0	2.5.0
file	5.11	5,39
file-libs	5.11	5,39
filesystem	3.2	3.14
findutils	4.5.11	4.8.0
fipscheck	1.4.1	
fipscheck-lib	1.4.1	
fonts-srpm-macros		2.0.5
freetype	2.8	
fstrm		0.6.1
fuse-libs	2.9.2	2.9.9
gawk	4.0.2	5.1.0
gdbm	1.13	
gdbm-libs		1,19

Pacote	AL2 AMI	AL2203 AMI
gdisk	0.8.10	1.0.8
generic-logos	18.0.0	
GeoIP	1.5.0	
gettext	0.19.8.1	0,21
gettext-libs	0.19.8.1	0,21
ghc-srpm-macros		1.5.0
glib2	2.56.1	2.74.7
glibc	2.26	2.34
glibc-all-langpacks	2.26	2.34
glibc-common	2.26	2.34
glibc-gconv-extra		2.34
glibc-locale-source	2.26	2.34
glibc-minimal-lang pack	2.26	
gmp	6.0.0	6.2.1
gnupg2	2.0.22	
gnupg2-minimal		2.3.7
gnutls		3.8.0
go-srpm-macros		3.2.0
gpgme	1.3.2	1.15.1
gpm-libs	1.20.7	1.20.7

Pacote	AL2 AMI	AL2203 AMI
grep	2.20	3.8
groff-base	1.22.2	1.22.4
grub2	2.06	
grub2-common	2.06	2.06
grub2-efi-aa64	2.06 (64 de março)	
grub2-efi-aa64-ec2	2.06 (64 de março)	2.06 (64 de março)
grub2-efi-aa64-modules	2.06 (março)	
grub2-efi-x64-ec2	2,06 (x86_64)	2,06 (x86_64)
grub2-pc	2,06 (x86_64)	
grub2-pc-modules	2.06 (março)	2.06
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8,28	8,40
gssproxy	0.7.0	0.8.4
gzip	1.5	1.12
hardlink	1.3	
hibagent	1.1.0	
hostname	3.13	3.23
hunspell	1.3.2	1.7.0
hunspell-en	0,20121024	0.20140811.1

Pacote	AL2 AMI	AL2203 AMI
hunspell-en-GB	0,20121024	0.20140811.1
hunspell-en-US	0,20121024	0.20140811.1
hunspell-filesystem		1.7.0
hwdata	0,252	0,384
info	5.1	6.7
inih		49
initscripts	9.49,47	10.09
iproute	5.10.0	6.10.0
iptables	1.8.4	
iptables-libs	1.8.4	
iputils	20180629	20210202
irqbalance	1.7.0	1.9.0
jansson	(2.10)	2.14
jbigkit-libs	2,0	
jemalloc		5.2.1
jitterentropy		3.4.1
jq		1.7.1
json-c	0,11	0,14
kbd	1.15.5	2.4.0
kbd-legacy	1.15.5	

Pacote	AL2 AMI	AL2203 AMI
kbd-misc	1.15.5	2.4.0
kernel	5.10.228	6.1.112
kernel-libbpf		6.1.112
kernel-livepatch-repo-s3		2023.6.20241031
kernel-srpm-macros		1,0
kernel-tools	5.10.228	6.1.112
keyutils	1.5.8	1.6.3
keyutils-libs	1.5.8	1.6.3
kmod	25	29
kmod-libs	25	29
kpartx	0.4.9	
kpatch-runtime	0.9.4	0.9.7
krb5-libs	1.15.1	1.21.3
langtable	0.0.31	
langtable-data	0.0.31	
langtable-python	0.0.31	
less	458	608
libacl	2.2.51	2.3.1
libaio	0.3.109	0.3.111
libarchive		3.7.4

Pacote	AL2 AMI	AL2203 AMI
libargon2		27 de dezembro de 2017
libassuan	2.1.0	2.5.5
libattr	2.4.46	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.30.2	2.37.4
libcap	2,54	2,48
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.42.9	1.46,5
libcomps		0.1.20
libconfig	1.4.9	1.7.2
libcroco	0.6.12	
libcrypt	2.26	
libcurl	8.3.0	
libcurl-minimal		8.5.0
libdaemon	0,14	
libdb	5.3.21	5.3.28
libdb-utils	5.3.21	
libdhash		0.5.0

Pacote	AL2 AMI	AL2203 AMI
libdnf		0.69,0
libdrm	2.4.97	
libdwarf	20130207 (x86_64)	
libeconf		0.4.0
libedit	3.0	3.1
libestr	0.1.9	
libev		4,33
libevent	2.0.21	2.1.12
libfastjson	0,99,4	
libfdisk	2.30.2	2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp	7.3.1	11.4.1
libgpg-error	1.12	1,42
libibverbs		48,0
libicu	50,2	
libidn	1,28	
libidn2	2.3.0	2.3.2

Pacote	AL2 AMI	AL2203 AMI
libini_config	1.3.1	1.3.1
libjpeg-turbo	2.0.90	
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libldb		2.6.2
libmaxminddb		1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.30.2	2.37.4
libnetfilter_conntrack	1.0.6	
libnfnetlink	1.0.1	
libnfsidmap	0.25	2.5.4
libnghttp2	1.41.0	1.59.0
libnl3	3.2.28	3.5.0
libnl3-cli	3.2.28	
libpath_utils	0.2.1	0.2.1
libpcap	1.5.3	1.10.1
libpciaccess	0,14 (x86_64)	
libpipeline	1.2.3	1.5.3

Pacote	AL2 AMI	AL2203 AMI
libpkgconf		1.8.0
libpng	1.5.13	
libpsl	0,21,5	0,21,1
libpwquality	1.2.3	1.4.4
libref_array	0.1.5	0.1.5
librepo		1.14.5
libreport-filesystem		2.15.2
libseccomp	2.5.2	2.5.3
libselinux	2,5	3.4
libselinux-utils	2,5	3.4
libsemanage	2,5	3.4
libsepol	2,5	3.4
libsigsegv		2.13
libsmartcols	2.30.2	2.37.4
libsolv		0.7.22
libss	1.42.9	1.46,5
libssh2	1.4.3	
libsss_certmap		2.9.4
libsss_idmap	1.16.5	2.9.4
libsss_nss_idmap	1.16.5	2.9.4

Pacote	AL2 AMI	AL2203 AMI
libsss_sudo		2.9.4
libstdc++	7.3.1	11.4.1
libstoragemgmt	1.6.1	1.9.4
libstoragemgmt-python	1.6.1	
libstoragemgmt-python-clibs	1.6.1	
libsysfs	2.1.0	
libtalloc		2.3.4
libtasn1	4.10	4.19.0
libtdb		1.4.7
libteam	1,27	
libtevent		0.13.0
libtextstyle		0,21
libtiff	4.0.3	
libtirpc	0.2.4	1.3.3
libunistring	0.9.3	0.9.10
libuser	0,60	0,63
libutempter	1.1.6	1.2.1
libuuid	2.30.2	2.37.4
libuv		1.47.0

Pacote	AL2 AMI	AL2203 AMI
libverto	0.2.5	0.3.2
libverto-libev		0.3.2
libverto-libevent	0.2.5	
libwebp	0.3.0	
libxcrypt		4.4.3
libxml2	2.9.1	2.10.4
libxml2-python	2.9.1	
libyaml	0.1.4	0.2.5
libzstd		1.5.5
linux-firmware-whence		20210208 (noarca)
lm_sensors-libs	3.4.0	3.6.0
lmdb-libs		0.9.29
logrotate	3.8.6	3.20.1
lsof	4,87	4.94.0
lua	5.1.4	
lua-libs		5.4.4
lua-srpm-macros		1
lvm2	2.02.187	
lvm2-libs	2.02.187	
lz4	1.7.5	

Pacote	AL2 AMI	AL2203 AMI
lz4-libs		1.9.4
make	3,82	
man-db	2.6.3	2.9.3
man-pages	3,53	5.10
man-pages-overrides	7.5.2	
mariadb-libs	5.5.68	
mdadm	4,0	
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)
mlocate	0,26	
mpfr		4.1.0
mtr	0.92	
nano	2.9.8	5,8
ncurses	6.0	6.2
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
nettle	2.7.1	3.8
net-tools	2,0	2.0
newt	0,52,15	0,52,21
newt-python	0,52,15	
nfs-utils	1.3.0	2.5.4

Pacote	AL2 AMI	AL2203 AMI
npth		1.6
nspr	4.35.0	4.35.0
nss	3.90,0	3.90,0
nss-pem	1.0.3	
nss-softokn	3.90,0	3.90,0
nss-softokn-freebl	3.90,0	3.90,0
nss-sysinit	3.90,0	3.90,0
nss-tools	3.90,0	
nss-util	3.90,0	3.90,0
ntsysv	1.7.4	1.15
numactl-libs	2.0.9	2.0.14
ocaml-srpm-macros		6
oniguruma		6.9.7.1
openblas-srpm-macros		2
openldap	2.4.44	2.4.57
openssh	7.4p1	8,7p1
openssh-clients	7.4p1	8,7p1
openssh-server	7.4p1	8,7p1
openssl	1,0,2 k	3.0.8
openssl-libs	1,0,2 k	3.0.8

Pacote	AL2 AMI	AL2203 AMI
openssl-pkcs11		0.4.12
os-prober	1,58	1,7
p11-kit	0.23.22	0.24.1
p11-kit-trust	0.23.22	0.24.1
package-notes-srpm-macros		0.4
pam	1.1.8	1.5.1
parted	3.1	3.4
passwd	0,79	0,80
pciutils	3.5.1	3.7.0
pciutils-libs	3.5.1	3.7.0
pcre	8,32	
pcre2	10,23	10,40
pcre2-syntax		10,40
perl	5.16.3	
perl-Carp	1,26	1,50
perl-Class-Struct		0,66
perl-constant	1,27	1,33
perl-DynaLoader		1,47
perl-Encode	2,51	3,15
perl-Errno		1,30

Pacote	AL2 AMI	AL2203 AMI
perl-Exporter	5,68	5,74
perl-Fcntl		1.13
perl-File-Basename		2,85
perl-File-Path	2.09	2,18
perl-File-stat		1,09
perl-File-Temp	0.23.01	0.231.100
perl-Filter	1,49	
perl-Getopt-Long	2,40	2,52
perl-Getopt-Std		1.12
perl-HTTP-Tiny	0,033	0,078
perl-if		0.60.800
perl-interpreter		5.32.1
perl-IO		1,43
perl-IPC-Open3		1,21
perl-libs	5.16.3	5.32.1
perl-macros	5.16.3	
perl-MIME-Base64		3,16
perl-mro		1,23
perl-overload		1,31
perl-overloading		0,02

Pacote	AL2 AMI	AL2203 AMI
perl-parent	0,225	0,238
perl-PathTools	3,40	3,78
perl-Pod-Escapes	1.04	1,07
perl-podlators	2.5.1	4.14
perl-Pod-Perldoc	3.20	3.28.01
perl-Pod-Simple	3,28	3,42
perl-Pod-Usage	1,63	2.01
perl-POSIX		1,94
perl-Scalar-List-Utils	1,27	1,56
perl-SelectSaver		1.02
perl-Socket	2.010	2.032
perl-srpm-macros		1
perl-Storable	2,45	3.21
perl-subs		1,03
perl-Symbol		1,08
perl-Term-ANSIColor		5.01
perl-Term-Cap		1.17
perl-Text-ParseWords	3,29	3,30
perl-Text-Tabs+Wrap		2021.07.26
perl-threads	1,87	

Pacote	AL2 AMI	AL2203 AMI
perl-threads-shared	1,43	
perl-Time-HiRes	1.9725	
perl-Time-Local	1.2300	1.300
perl-vars		1,05
pinentry	0.8.1	
pkgconf		1.8.0
pkgconfig	0.27.1	
pkgconf-m4		1.8.0
pkgconf-pkg-config		1.8.0
plymouth	0.8.9	
plymouth-core-libs	0.8.9	
plymouth-scripts	0.8.9	
pm-utils	1.4.1	
policycoreutils	2,5	3.4
policycoreutils-python-utils		3.4
popt	1.13	1,18
postfix	2.10.1	
procps-ng	3.3.10	3.3.17
protobuf-c		1.4.1
psacct	6.6.1	6.6.4

Pacote	AL2 AMI	AL2203 AMI
psmisc	22,20	23,4
pth	2.0.7	
publicsuffix-list-dafsa	20240208	2024/02/12
pygpgme	0.3	
pyliblzma	0.5.3	
pystache	0.5.3	
python	2.7.18	
python2-botocore	1.18.6	
python2-colorama	0.3.9	
python2-cryptography	1.7.2	
python2-dateutil	2.6.1	
python2-futures	3.0.5	
python2-jmespath	0.9.3	
python2-jsonschema	2.5.1	
python2-oauthlib	2.0.1	
python2-pyasn1	0.1.9	
python2-rpm	4.11.3	
python2-rsa	3.4.1	
python2-s3transfer	0.3.3	
python2-setuptools	41.2.0	

Pacote	AL2 AMI	AL2203 AMI
python2-six	1.11.0	
python3	3.7.16	3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-daemon	2.2.3	2.3.0
python3-dateutil		2.8.1
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils	0,14	0,16
python3-gpg		1.15.1
python3-hawkey		0.69,0

Pacote	AL2 AMI	AL2203 AMI
python3-idna		(2.10)
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1,21
python3-jsonpointer		2,0
python3-jsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69,0
python3-libs	3.7.16	3.9.16
python3-libselinux		3.4
python3-libsemanage		3.4
python3-libstorage mgmt		1.9.4
python3-lockfile	0.11.0	0.12.2
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip	20.2.2	
python3-pip-wheel		21.3.1
python3-ply		3.11

Pacote	AL2 AMI	AL2203 AMI
python3-policycoreutils		3.4
python3-prettytable		0.7.2
python3-prompt-toolkit		3.0.24
python3-pycparser		2.20
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pystache	0.5.4	
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2
python3-setools		4.4.1
python3-setuptools	49.1.3	59.6.0
python3-setuptools-wheel		59.6.0
python3-simplejson	3.2.0	

Pacote	AL2 AMI	AL2203 AMI
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
python-babel	0.9.6	
python-backports	1,0	
python-backports-ssl_match_hostname	3.5.0.1	
python-cffi	1.6.0	
python-chardet	2.2.1	
python-chevron		0.13.1
python-configobj	4.7.2	
python-daemon	1.6	
python-devel	2.7.18	
python-docutils	0,12	
python-enum34	1.0.4	
python-idna	2.4	
python-iniparse	0.4	
python-ipaddress	1.0.16	
python-jinja2	2.7.2	
python-jsonpatch	1.2	

Pacote	AL2 AMI	AL2203 AMI
python-jsonpointer	1.9	
python-jwcrypto	0.4.2	
python-kitchen	1.1.1	
python-libs	2.7.18	
python-lockfile	0.9.1	
python-markupsafe	0,11	
python-pillow	2.0.0	
python-ply	3.4	
python-pycparser	2.14	
python-pycurl	7.19.0	
python-repoze-lru	0.4	
python-requests	2.6.0	
python-simplejson	3.2.0	
python-srpm-macros		3.9
python-urlgrabber	3.10	
python-urllib3	1.25.9	
pyxattr	0.5.1	
PyYAML	3.10	
qrencode-libs	3.4.1	
quota	4.01	4.06

Pacote	AL2 AMI	AL2203 AMI
quota-nls	4.01	4.06
rdate	1.4	
readline	6.2	8.1
rng-tools	6.8	6.14
rootfiles	8.1	8.1
rpcbind	0.2.0	1.2.6
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit	4.11.3	4.16.1.3
rpm-sign-libs		4.16.1.3
rsync	3.1.2	3.2.6
rsyslog	8.24,0	
rust-srpm-macros		21
sbsigntools		0.9.4
scl-utils	20130529	
screen	4.1.0	4.8.0
sed	4.2.2	4.8
selinux-policy	3.13.1	38.1.45

Pacote	AL2 AMI	AL2203 AMI
selinux-policy-targeted	3.13.1	38.1.45
setserial	2,17	
setup	2.8.71	2.13.7
setuptool	1.19.11	
sgpio	1.2.0.10	
shadow-utils	4.1.5.1	4,9
shared-mime-info	1.8	
slang	2.2.4	2.3.2
sqlite	3.7.17	
sqlite-libs		3.40,0
sssd-client	1.16.5	2.9.4
sssd-common		2.9.4
sssd-kcm		2.9.4
sssd-nfs-idmap		2.9.4
strace	4.26	6.8
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1,0
sysstat	10.1.5	12.5.6
systemd	219	252,23
systemd-libs	219	252,23

Pacote	AL2 AMI	AL2203 AMI
systemd-networkd		252,23
systemd-pam		252,23
systemd-resolved		252,23
systemd-sysv	219	
systemd-udev		252,23
system-release	2	2023.6.20241031
systemtap-runtime	4.5	4.8
sysvinit-tools	2,88	
tar	1,26	1,34
tbb		2020.3
tcp_wrappers	7.6	
tcp_wrappers-libs	7.6	
tcpdump	4.9.2	4.99.1
tcsh	6.18.01	24.6.07
teamd	1,27	
time	1,7	1.9
traceroute	2.0.22	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	1.1.2	2.2

Pacote	AL2 AMI	AL2203 AMI
usermode	1,111	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.30.2	2.37.4
util-linux-core		2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
virt-what	1,18	
wget	1.14	1.21.3
which	2.20	2.21
words	3.0	3.0
xfsdump	3.1.8	3.1.11
xfsprogs	5.0.0	5.18.0
xxd	9.0.2153	9.0.2153
xxhash-libs		0.8.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5

Pacote	AL2 AMI	AL2203 AMI
yajl	2.0.4	
yum	3.4.3	4.14.0
yum-langpacks	0.4.2	
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
yum-utils	1.1.31	
zip	3.0	3.0
zlib	1.2.7	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

Comparando pacotes instalados no Amazon Linux 2 e no Amazon Linux 2023 Minimal AMIs

Uma comparação do RPMs presente no Amazon Linux 2 e no AL2 023 Minimal AMIs.

Pacote	AL2 Mínimo	AL2023 Mínimo
acl	2.2.51	
alternatives		1.15
amazon-chrony-config		4.3

Pacote	AL2 Mínimo	AL2023 Mínimo
amazon-ec2-net-utils		2.5.1
amazon-linux-extras	2.0.3	
amazon-linux-repo-s3		2023.6.20241031
amazon-linux-sb-keys		2023.1
amd-ucode-firmware	2020421 (noarca)	20210208 (noarca)
audit	2.8.1	3.0.6
audit-libs	2.8.1	3.0.6
authconfig	6.2.8	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
bind-export-libs	9.11.4	
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.68	2023.2.68
checkpolicy		3.4
chkconfig	1.7.4	
chrony	4.2	4.3
cloud-init	19.3	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart	0,31	0,31

Pacote	AL2 Mínimo	AL2023 Mínimo
coreutils	8.22	8,32
coreutils-common		8,32
cpio	2.12	2.13
cracklib	2.9.0	2.9.6
cracklib-dicts	2.9.0	2.9.6
cronie	1.4.11	
cronie-anacron	1.4.11	
crontabs	1.11	
crypto-policies		2020428
cryptsetup-libs	1.7.4	2.6.1
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	2.1.27
dbus	1.10.24	1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.10.24	1.12.28
device-mapper	1.02.170	1.02.185
device-mapper-libs	1.02.170	1.02.185
dhclient	4.2.5	

Pacote	AL2 Mínimo	AL2023 Mínimo
dhcp-common	4.2.5	
dhcp-libs	4.2.5	
diffutils	3.3	3.8
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dracut	033	055
dracut-config-ec2	2,0	3.0
dracut-config-generic	033	055
e2fsprogs	1.42.9	1.46,5
e2fsprogs-libs	1.42.9	1.46,5
ec2-utils	1.2	2.2.0
efibootmgr	15 (março de 64)	
efi-filesystem		5
efivar		38
efivar-libs	31 (março 64)	38

Pacote	AL2 Mínimo	AL2023 Mínimo
elfutils-default-yama-scope	0,176	0.188
elfutils-libelf	0,176	0.188
elfutils-libs	0,176	0.188
expat	2.1.0	2.5.0
file	5.11	5,39
file-libs	5.11	5,39
filesystem	3.2	3.14
findutils	4.5.11	4.8.0
fipscheck	1.4.1	
fipscheck-lib	1.4.1	
freetype	2.8	
fuse-libs	2.9.2	2.9.9
gawk	4.0.2	5.1.0
gdbm	1.13	
gdbm-libs		1,19
gdisk	0.8.10	1.0.8
gettext	0.19.8.1	0,21
gettext-libs	0.19.8.1	0,21
glib2	2.56.1	2.74.7
glibc	2.26	2.34

Pacote	AL2 Mínimo	AL2023 Mínimo
glibc-all-langpacks	2.26	2.34
glibc-common	2.26	2.34
glibc-locale-source	2.26	2.34
glibc-minimal-langpack	2.26	
gmp	6.0.0	6.2.1
gnupg2	2.0.22	
gnupg2-minimal		2.3.7
gnutls		3.8.0
gpgme	1.3.2	1.15.1
grep	2.20	3.8
groff-base	1.22.2	1.22.4
grub2	2.06	
grub2-common	2.06	2.06
grub2-efi-aa64	2.06 (64 de março)	
grub2-efi-aa64-ec2	2.06 (64 de março)	2.06 (64 de março)
grub2-efi-aa64-modules	2.06 (março)	
grub2-efi-x64-ec2	2,06 (x86_64)	2,06 (x86_64)
grub2-pc	2,06 (x86_64)	
grub2-pc-modules	2.06 (março)	2.06

Pacote	AL2 Mínimo	AL2023 Mínimo
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8,28	8,40
gzip	1.5	1.12
hardlink	1.3	
hostname	3.13	3.23
hwdata		0,384
info	5.1	
inih		49
initscripts	9.49,47	10.09
iproute	5.10.0	6.10.0
iptables	1.8.4	
iptables-libs	1.8.4	
iputils	20180629	20210202
irqbalance	1.7.0	1.9.0
jansson		2.14
jitterentropy		3.4.1
jq		1.7.1
json-c		0,14
kbd		2.4.0

Pacote	AL2 Mínimo	AL2023 Mínimo
kbd-misc		2.4.0
kernel	4.14.355	6.1.112
kernel-libbpf		6.1.112
kernel-livepatch-repo-s3		2023.6.20241031
keyutils-libs	1.5.8	1.6.3
kmod	25	29
kmod-libs	25	29
kpartx	0.4.9	
krb5-libs	1.15.1	1.21.3
less	458	608
libacl	2.2.51	2.3.1
libarchive		3.7.4
libargon2		27 de dezembro de 2017
libassuan	2.1.0	2.5.5
libattr	2.4.46	2.5.1
libblkid	2.30.2	2.37.4
libcap	2,54	2,48
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcom_err	1.42.9	1.46,5

Pacote	AL2 Mínimo	AL2023 Mínimo
libcomps		0.1.20
libcroco	0.6.12	
libcrypt	2.26	
libcurl	8.3.0	
libcurl-minimal		8.5.0
libdb	5.3.21	5.3.28
libdb-utils	5.3.21	
libdnf		0.69,0
libeconf		0.4.0
libedit	3.0	3.1
libestr	0.1.9	
libfastjson	0,99,4	
libfdisk	2.30.2	2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp	7.3.1	11.4.1
libgpg-error	1.12	1,42
libicu	50,2	

Pacote	AL2 Mínimo	AL2023 Mínimo
libidn	1,28	
libidn2	2.3.0	2.3.2
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libmetalink	0.1.3	
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.30.2	2.37.4
libnetfilter_conntrack	1.0.6	
libnfnetlink	1.0.1	
libnghttp2	1.41.0	1.59.0
libpcap	1.5.3	
libpipeline	1.2.3	1.5.3
libpng	1.5.13	
libpsl	0,21,5	0.21.1
libpwquality	1.2.3	1.4.4
librepo		1.14.5
libreport-filesystem		2.15.2
libseccomp	2.5.2	2.5.3
libselinux	2,5	3.4

Pacote	AL2 Mínimo	AL2023 Mínimo
libselinux-utils	2,5	3.4
libsemanage	2,5	3.4
libsepol	2,5	3.4
libsigsegv		2.13
libsmartcols	2.30.2	2.37.4
libsolv		0.7.22
libss	1.42.9	1.46,5
libssh2	1.4.3	
libstdc++	7.3.1	11.4.1
libsysfs	2.1.0	
libtasn1	4.10	4.19.0
libtextstyle		0,21
libunistring	0.9.3	0.9.10
libuser	0,60	0,63
libutempter	1.1.6	1.2.1
libuuid	2.30.2	2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.3
libxml2	2.9.1	2.10.4
libyaml	0.1.4	0.2.5

Pacote	AL2 Mínimo	AL2023 Mínimo
libzstd		1.5.5
linux-firmware-whence		20210208 (noarca)
logrotate	3.8.6	3.20.1
lua	5.1.4	
lua-libs		5.4.4
lz4	1.7.5	
lz4-libs		1.9.4
make	3,82	
man-db	2.6.3	2.9.3
mariadb-libs	5.5.68	
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)
mpfr		4.1.0
ncurses	6.0	6.2
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
nettle	2.7.1	3.8
net-tools	2,0	2.0
newt	0,52,15	
newt-python	0,52,15	
npth		1.6

Pacote	AL2 Mínimo	AL2023 Mínimo
nspr	4.35.0	
nss	3.90,0	
nss-pem	1.0.3	
nss-softokn	3.90,0	
nss-softokn-freebl	3.90,0	
nss-sysinit	3.90,0	
nss-tools	3.90,0	
nss-util	3.90,0	
numactl-libs	2.0.9	2.0.14
oniguruma		6.9.7.1
openldap	2.4.44	2.4.57
openssh	7.4p1	8,7p1
openssh-clients	7.4p1	8,7p1
openssh-server	7.4p1	8,7p1
openssl	1,0,2 k	3.0.8
openssl-libs	1,0,2 k	3.0.8
openssl-pkcs11		0.4.12
os-prober	1,58	1,7
p11-kit	0.23.22	0.24.1
p11-kit-trust	0.23.22	0.24.1

Pacote	AL2 Mínimo	AL2023 Mínimo
pam	1.1.8	1.5.1
passwd	0,79	0,80
pciutils		3.7.0
pciutils-libs		3.7.0
pcre	8,32	
pcre2	10,23	10h40
pcre2-syntax		10h40
pinentry	0.8.1	
pkgconfig	0.27.1	
policycoreutils	2,5	3.4
popt	1.13	1,18
postfix	2.10.1	
procps-ng	3.3.10	3.3.17
psmisc	22,20	23,4
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0.5.3	
python	2.7.18	
python2-cryptography	1.7.2	

Pacote	AL2 Mínimo	AL2023 Mínimo
python2-jsonschema	2.5.1	
python2-oauthlib	2.0.1	
python2-pyasn1	0.1.9	
python2-rpm	4.11.3	
python2-setuptools	41.2.0	
python2-six	1.11.0	
python3		3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-dateutil		2.8.1
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0

Pacote	AL2 Mínimo	AL2023 Mínimo
python3-dnf-plugins-core		4.3.0
python3-docutils		0,16
python3-gpg		1.15.1
python3-hawkey		0.69,0
python3-idna		(2.10)
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1,21
python3-jsonpointer		2,0
python3-jsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69,0
python3-libs		3.9.16
python3-libselinux		3.4
python3-libsemanage		3.4
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3.11

Pacote	AL2 Mínimo	AL2023 Mínimo
python3-policycoreutils		3.4
python3-prettytable		0.7.2
python3-prompt-toolkit		3.0.24
python3-pycparser		2.20
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2
python3-setools		4.4.1
python3-setuptools		59.6.0
python3-setuptools-wheel		59.6.0
python3-six		1.15.0
python3-systemd		235

Pacote	AL2 Mínimo	AL2023 Mínimo
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
python-babel	0.9.6	
python-backports	1,0	
python-backports-ssl_match_hostname	3.5.0.1	
python-cffi	1.6.0	
python-chardet	2.2.1	
python-configobj	4.7.2	
python-devel	2.7.18	
python-enum34	1.0.4	
python-idna	2.4	
python-iniparse	0.4	
python-ipaddress	1.0.16	
python-jinja2	2.7.2	
python-jsonpatch	1.2	
python-jsonpointer	1.9	
python-jwcrypto	0.4.2	
python-libs	2.7.18	
python-markupsafe	0,11	
python-ply	3.4	

Pacote	AL2 Mínimo	AL2023 Mínimo
python-pycparser	2.14	
python-pycurl	7.19.0	
python-repoze-lru	0.4	
python-requests	2.6.0	
python-urlgrabber	3.10	
python-urllib3	1.25.9	
pyxattr	0.5.1	
PyYAML	3.10	
qrencode-libs	3.4.1	
readline	6.2	8.1
rng-tools	6.8	6.14
rootfiles	8.1	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit	4.11.3	4.16.1.3
rpm-sign-libs		4.16.1.3
rsyslog	8.24,0	
sbsigntools		0.9.4

Pacote	AL2 Mínimo	AL2023 Mínimo
sed	4.2.2	4.8
selinux-policy	3.13.1	38.1.45
selinux-policy-targeted	3.13.1	38.1.45
setup	2.8.71	2.13.7
shadow-utils	4.1.5.1	4,9
shared-mime-info	1.8	
slang	2.2.4	
sqlite	3.7.17	
sqlite-libs		3.40,0
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1,0
systemd	219	252,23
systemd-libs	219	252,23
systemd-networkd		252,23
systemd-pam		252,23
systemd-resolved		252,23
systemd-sysv	219	
systemd-udev		252,23
system-release	2	2023.6.20241031
sysvinit-tools	2,88	

Pacote	AL2 Mínimo	AL2023 Mínimo
tar	1,26	1,34
tcp_wrappers-libs	7.6	
tzdata	2024a	2024a
update-motd	1.1.2	2.2
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.30.2	2.37.4
util-linux-core		2.37.4
vim-data	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
which	2.20	2.21
xfsprogs	5.0.0	5.18.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
zlib	1.2.7	1.2.11
zram-generator		1.1.2

Pacote	AL2 Mínimo	AL2023 Mínimo
zram-generator-defaults		1.1.2
zstd		1.5.5

Compare de comparação de pacotes instalados em imagens de contêiner de base do Amazon Linux 2023 e do Amazon Linux 2023

Uma comparação do RPMs presente nas imagens de contêineres básicos do Amazon Linux 2 e AL2 023.

Pacote	AL2 Contêiner	AL2Recipiente 023
alternatives		1.15
amazon-linux-extras	2.0.3	
amazon-linux-repo-cdn		2023.6.20241031
audit-libs		3.0.6
basesystem	10.0	11
bash	4.2.46	5.2.15
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.68	2023.2.68
chkconfig	1.7.4	
coreutils	8.22	
coreutils-single		8,32
cpio	2,12	

Pacote	AL2 Contêiner	AL2Recipiente 023
crypto-policies		2020428
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	
diffutils	3.3	
dnf		4.14.0
dnf-data		4.14.0
elfutils-default-yama-scope		0.188
elfutils-libelf	0,176	0.188
elfutils-libs		0.188
expat	2.1.0	2.5.0
file-libs	5.11	5,39
filesystem	3.2	3,14
findutils	4.5.11	
gawk	4.0.2	5.1.0
gdbm	1.13	
gdbm-libs		1,19
glib2	2.56.1	2.74.7
glibc	2.26	2.34
glibc-common	2.26	2.34

Pacote	AL2 Contêiner	AL2Recipiente 023
glibc-langpack-en	2.26	
glibc-minimal-langpack	2.26	2.34
gmp	6.0.0	6.2.1
gnupg2	2.0.22	
gnupg2-minimal		2.3.7
gpgme	1.3.2	1.15.1
grep	2.20	3.8
info	5.1	
json-c		0,14
keyutils-libs	1.5.8	1.6.3
krb5-libs	1.15.1	1.21.3
libacl	2.2.51	2.3.1
libarchive		3.7.4
libassuan	2.1.0	2.5.5
libattr	2.4.46	2.5.1
libblkid	2.30.2	2.37.4
libcap	2,54	2,48
libcap-ng		0.8.2
libcom_err	1.42.9	1.46,5
libcomps		0.1.20

Pacote	AL2 Contêiner	AL2Recipiente 023
libcrypt	2.26	
libcurl	8.3.0	
libcurl-minimal		8.5.0
libdb	5.3.21	
libdb-utils	5.3.21	
libdnf		0.69,0
libffi	3.0.13	3.4.4
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.12	1,42
libidn2	2.3.0	2.3.2
libmetalink	0.1.3	
libmodulemd		2.13.0
libmount	2.30.2	2.37.4
libnghttp2	1.41.0	1.59.0
libpsl	0,21,5	0,21,1
librepo		1.14.5
libreport-filesystem		2.15.2
libselinux	2,5	3.4

Pacote	AL2 Contêiner	AL2Recipiente 023
libsepol	2,5	3.4
libsigsegv		2.13
libsmartcols		2.37.4
libsolv		0.7.22
libssh2	1.4.3	
libstdc++	7.3.1	11.4.1
libtasn1	4.10	4.19.0
libunistring	0.9.3	0.9.10
libuuid	2.30.2	2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.3
libxml2	2.9.1	2.10.4
libyaml		0.2.5
libzstd		1.5.5
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
mpfr		4.1.0
ncurses	6.0	
ncurses-base	6.0	6.2

Pacote	AL2 Contêiner	AL2Recipiente 023
ncurses-libs	6.0	6.2
npth		1.6
nspr	4.35.0	
nss	3.90,0	
nss-pem	1.0.3	
nss-softokn	3.90,0	
nss-softokn-freebl	3.90,0	
nss-sysinit	3.90,0	
nss-tools	3.90,0	
nss-util	3.90,0	
openldap	2.4.44	
openssl-libs	1,0,2 k	3.0.8
p11-kit	0.23.22	0.24.1
p11-kit-trust	0.23.22	0.24.1
pcre	8,32	
pcre2		10h40
pcre2-syntax		10h40
pinentry	0.8.1	
popt	1.13	1,18
pth	2.0.7	

Pacote	AL2 Contêiner	AL2Recipiente 023
publicsuffix-list-dafsa	20240208	2024/02/12
pygpgme	0.3	
pyliblzma	0.5.3	
python	2.7.18	
python2-rpm	4.11.3	
python3		3.9.16
python3-dnf		4.14.0
python3-gpg		1.15.1
python3-hawkey		0.69,0
python3-libcomps		0.1.20
python3-libdnf		0.69,0
python3-libs		3.9.16
python3-pip-wheel		21.3.1
python3-rpm		4.16.1.3
python3-setuptools-wheel		59.6.0
python-iniparse	0.4	
python-libs	2.7.18	
python-pycurl	7.19.0	
python-urlgrabber	3.10	

Pacote	AL2 Contêiner	AL2Recipiente 023
pyxattr	0.5.1	
readline	6.2	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-sign-libs		4.16.1.3
sed	4.2.2	4.8
setup	2.8.71	2.13.7
shared-mime-info	1.8	
sqlite	3.7.17	
sqlite-libs		3.40,0
system-release	2	2023.6.20241031
tzdata	2024a	2024a
vim-data	9.0.2153	
vim-minimal	9.0.2153	
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-ovl	1.1.31	
yum-plugin-priorities	1.1.31	

Pacote	AL2 Contêiner	AL2Recipiente 023
`zlib`	1.2.7	1.2.11

Comparando AL1 e AL2 023

Os tópicos a seguir descrevem as principais diferenças entre AL1 e AL2 023 que ainda não foram abordadas pela comparação com AL2.

> **Note**
>
> AL1 atingiu seu end-of-life (EOL) em 31 de dezembro de 2023 e não receberá nenhuma atualização de segurança ou correção de bugs a partir de 1º de janeiro de 2024. Para obter mais informações sobre AL1 EOL e suporte de manutenção, consulte a postagem do blog Update on Amazon Linux AMI end-of-life. Recomendamos que você atualize os aplicativos para a versão AL2 023, que inclui suporte de longo prazo até 2028.

Tópicos

Suporte para cada versão

Para AL2 2013, oferecemos cinco anos de suporte a partir da data de lançamento. AL1 encerrou o suporte padrão em 31 de dezembro de 2020 e encerrou o suporte de manutenção em 31 de dezembro de 2023.

Para obter mais informações, consulte Cadência de lançamento.

`systemd` substitui `upstart` como sistema `init`

AL2 `upstart`Foi substituído por `systemd` as the `init` system. AL2O 023 também utiliza `systemd` como seu `init` sistema, adotando ainda mais novos recursos e funcionalidades do. `systemd`

Python 2.6 e 2.7 foram substituídos pelo Python 3

Embora tenha AL1 marcado o Python 2.6 como EOL na versão 2018.03, os pacotes ainda estavam disponíveis nos repositórios para instalação. AL2 fornecido com o Python 2.7 como a primeira versão compatível do Python, AL2 e o 023 completa a transição para o Python 3. Nenhuma versão 2.x do Python está incluída nos repositórios 023. AL2

Para obter mais informações sobre Python no Amazon Linux, consulte Python em AL2 023.

OpenJDK 8 como o JDK mais antigo

AL2O 023 é fornecido com o Amazon Corretto como o Java Development Kit (JDK) padrão (e único). Todos Java pacotes baseados em AL2 023 são construídos com Amazon Corretto 17.

Em AL1, o OpenJDK 1.6.0 `java-1.6.0-openjdk` () tornou-se EOL com a primeira versão 2018.03, e o OpenJDK 1.7.0 () tornou-se EOL em meados de 2020, embora ambas as versões estivessem disponíveis nos repositórios. `java-1.7.0-openjdk` AL1 A versão mais antiga do OpenJDK disponível em AL2 023 é o OpenJDK 8, fornecido pela Amazon Corretto 8.

AL2023 alterações no kernel do Amazon Linux (1) AL1

Kernel Live Patching

Tanto o AL2 023 quanto o AL2 adicionam suporte para a funcionalidade de correção ao vivo do kernel. Isso permite corrigir vulnerabilidades de segurança críticas e importantes no kernel Linux sem reinicialização ou tempo de inatividade. Para obter mais informações, consulte Atualização do Kernel Live em 023 AL2.

Suporte ao sistema de arquivos do kernel

Houve várias mudanças nos sistemas de arquivos nos quais o kernel AL1 suportará a montagem, junto com mudanças nos esquemas de particionamento que o kernel analisará.

Opção do **CONFIG**	AL1/4.14/ x86_64	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_AFS_FS	m	n	n	n	n
CONFIG_AF_RXRPC	m	n	n	n	n
CONFIG_BSD_DISKLABEL	y	n	n	n	n
CONFIG_CRAMFS	m	n	n	n	n
CONFIG_CRAMFS_BLOCKDEV	N/D	N/D	N/D	N/D	N/D
CONFIG_DM_CLONE	N/D	n	n	n	n
CONFIG_DM_ERA	n	n	n	n	n
CONFIG_DM_INTEGRITY	m	m	m	m	m
CONFIG_DM_LOG_WRITES	n	m	m	m	m
CONFIG_DM_SWITCH	n	n	n	n	n
CONFIG_DM_VERITY	n	n	n	n	n

Opção do **CONFIG**	AL1/4.14/ x86_64	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_ECRYPT_FS	m	n	n	n	n
CONFIG_EXFAT_FS	N/D	m	m	m	m
CONFIG_EXT2_FS	m	n	n	n	n
CONFIG_EXT3_FS	m	n	n	n	n
CONFIG_GFS2_FS	n	n	n	n	n
CONFIG_HFSPLUS_FS	m	n	n	n	n
CONFIG_HFS_FS	m	n	n	n	n
CONFIG_JFS_FS	n	n	n	n	n
CONFIG_LDM_PARTITION	y	n	n	n	n
CONFIG_MAC_PARTITION	y	n	n	n	n
CONFIG_NFS_V2	m	n	n	n	n

Opção do **CONFIG**	AL1/4.14/ x86_64	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_NTFS_FS	m	n	n	n	n
CONFIG_ROMFS_FS	m	n	n	n	n
CONFIG_SOLARIS_X86_PARTITION	y	n	n	n	n
CONFIG_SQUASHFS_ZSTD	y	y	y	y	y
CONFIG_SUN_PARTITION	y	n	n	n	n

Alterações na configuração do kernel com foco na segurança

Opção do **CONFIG**	AL1/4.14/ x86_64	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_BUG_ON_DATA_CORRUPTION	y	y	y	y	y
CONFIG_DEFAULT_MMAP_MIN_ADDR	4096	65536	65536	65536	65536

Opção do **CONFIG**	AL1/4.14/ x86_64	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_DEVMEM	y	n	n	n	n
CONFIG_DEVPORT	y	n	n	n	n
CONFIG_FORTIFY_SOURCE	y	y	y	y	y
CONFIG_HARDENED_USERCOPY_FALLBACK	N/D	N/D	N/D	N/D	N/D
CONFIG_INIT_ON_ALLOC_DEFAULT_ON	N/D	n	n	n	n
CONFIG_INIT_ON_FREE_DEFAULT_ON	N/D	n	n	n	n
CONFIG_IOMMU_DEFAULT_DMA_STRICT	N/D	n	n	n	n
CONFIG_LDISC_AUTOLOAD	y	n	n	n	n

Opção do **CONFIG**	AL1/4.14/ x86_64	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_SCHED_CORE	N/D	N/D	y	N/D	y
CONFIG_SCHED_STACK_END_CHECK	y	y	y	y	y
CONFIG_SECURITY_DMESG_RESTRICT	n	y	y	y	y
CONFIG_SECURITY_SELINUX_DISABLE	y	n	n	N/D	N/D
CONFIG_SHUFFLE_PAGE_ALLOCATOR	N/D	y	y	y	y
CONFIG_SLAB_FREELIST_HARDENED	y	y	y	y	y
CONFIG_SLAB_FREELIST_RANDOM	n	y	y	y	y

Outras alterações na configuração do kernel

Opção do **CONFIG**	AL1/4.14/ x86_64	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_HZ	250	100	100	100	100
CONFIG_NR_CPUS	8192	4096	8192	4096	8192
CONFIG_PANIC_ON_OOPS	n	y	y	y	y
CONFIG_PANIC_ON_OOPS_VALUE	0	1	1	1	1
CONFIG_PPP	m	n	n	n	n
CONFIG_SLIP	m	n	n	n	n
CONFIG_XEN_PV	y	N/D	n	N/D	n

Comparando pacotes instalados no Amazon Linux 1 (AL1) e no Amazon Linux 2023 AMIs

Uma comparação do RPMs presente no padrão AL1 AMIs e AL2 023.

Pacote	AL1 AMI	AL2203 AMI
acl	2.2.49	2.3.1
acpid	2.0.19	2.0.32
alsa-lib	1.0.22	

Pacote	AL1 AMI	AL2203 AMI
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-repo-s3		2023.6.20241031
amazon-linux-sb-keys		2023.1
amazon-rpm-config		228
amazon-ssm-agent	3.2.222.0	3.3.987.0
amd-ucode-firmware		20210208
at	3.1.10	3.1.23
attr	2.4.46	2.5.1
audit	2.6.5	3.0.6
audit-libs	2.6.5	3.0.6
authconfig	6.2.8	
aws-amitools-ec2	1.5.13	
aws-cfn-bootstrap	1.4	2,0
aws-cli	1.18.107	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
bash-completion		2.11

Pacote	AL1 AMI	AL2203 AMI
bc	1.06.95	1.07.1
bind-libs	9.8.2	9.18.28
bind-license		9.18.28
bind-utils	9.8.2	9.18.28
binutils	2.27	2,39
boost-filesystem		1.75.0
boost-system		1.75.0
boost-thread		1.75.0
bzip2	1.0.6	1.0.8
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.62	2023.2.68
c-ares		1.19.1
checkpolicy	2.1.10	3.4
chkconfig	1.3.49.3	1.15
chrony		4.3
cloud-disk-utils	0,27	
cloud-init	0.7.6	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart		0,31
copy-jdk-configs	3.3	

Pacote	AL1 AMI	AL2203 AMI
coreutils	8.22	8,32
coreutils-common		8,32
cpio	(2.10)	2.13
cracklib	2.8.16	2.9.6
cracklib-dicts	2.8.16	2.9.6
cronie	1.4.4	
cronie-anacron	1.4.4	
crontabs	1.10	1.11
crypto-policies		2020428
crypto-policies-scripts		2020428
cryptsetup	1.6.7	2.6.1
cryptsetup-libs	1.6.7	2.6.1
curl	7.61.1	
curl-minimal		8.5.0
cyrus-sasl	2.1.23	
cyrus-sasl-lib	2.1.23	2.1.27
cyrus-sasl-plain	2.1.23	2.1.27
dash	0.5.5.1	
db4	4.7.25	
db4-utils	4.7.25	

Pacote	AL1 AMI	AL2203 AMI
dbus	1.6.12	1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.6.12	1.12.28
dejavu-fonts-common	2.33	
dejavu-sans-fonts	2.33	
dejavu-serif-fonts	2.33	
device-mapper	1.02.135	1.02.185
device-mapper-event	1.02.135	
device-mapper-event-libs	1.02.135	
device-mapper-libs	1.02.135	1.02.185
device-mapper-persistent-data	0.6.3	
dhclient	4.1.1	
dhcp-common	4.1.1	
diffutils	3.3	3.8
dmraid	1.0.0.rc16	
dmraid-events	1.0.0.rc16	
dnf		4.14.0
dnf-data		4.14.0

Pacote	AL1 AMI	AL2203 AMI
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dnf-utils		4.3.0
dosfstools		4.2
dracut	004	055
dracut-config-ec2		3.0
dracut-config-generic		055
dracut-modules-growroot	0.20	
dump	0.4	
dwz		0,14
dyninst		10.2.1
e2fsprogs	1.43.5	1.46,5
e2fsprogs-libs	1.43.5	1.46,5
ec2-hibinit-agent	1.0.0	1.0.8
ec2-instance-connect		1.1
ec2-instance-connect-selinux		1.1

Pacote	AL1 AMI	AL2203 AMI
ec2-net-utils	0.7	
ec2-utils	0.7	2.2.0
ed	1.1	1.14.2
efi-filesystem		5
efi-srpm-macros		5
efivar		38
efivar-libs		38
elfutils-debuginfod-client		0.188
elfutils-default-yama-scope		0.188
elfutils-libelf	0,168	0.188
elfutils-libs		0.188
epel-release	6	
ethtool	3,15	5,15
expat	2.1.0	2.5.0
file	5,37	5,39
file-libs	5,37	5,39
filesystem	2.4.30	3,14
findutils	4.4.2	4.8.0
fipscheck	1.3.1	

Pacote	AL1 AMI	AL2203 AMI
fipscheck-lib	1.3.1	
fontconfig	2.8.0	
fontpackages-filesystem	1,41	
fonts-srpm-macros		2.0.5
freetype	2.3.11	
fstrm		0.6.1
fuse-libs	2.9.4	2.9.9
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1,19
gdisk	0.8.10	1.0.8
generic-logos	17.0.0	
get_reference_source	1.2	
gettext		0,21
gettext-libs		0,21
ghc-srpm-macros		1.5.0
giflib	4.1.6	
glib2	2.36.3	2.74.7
glibc	2,17	2.34
glibc-all-langpacks		2.34

Pacote	AL1 AMI	AL2203 AMI
glibc-common	2,17	2.34
glibc-gconv-extra		2.34
glibc-locale-source		2.34
gmp	6.0.0	6.2.1
gnupg2	2.0.28	
gnupg2-minimal		2.3.7
gnutls		3.8.0
go-srpm-macros		3.2.0
gpgme	1.4.3	1.15.1
gpm-libs	1.20.6	1.20.7
grep	2.20	3.8
groff	1.22.2	
groff-base	1.22.2	1.22.4
grub	0,97	
grub2-common		2.06
grub2-efi-x64-ec2		2.06
grub2-pc-modules		2.06
grub2-tools		2.06
grub2-tools-minimal		2.06
grubby	7.0.15	8,40

Pacote	AL1 AMI	AL2203 AMI
gssproxy		0.8.4
gzip	1.5	1.12
hesiod	3.1.0	
hibagent	1.0.0	
hmaccalc	0.9.12	
hostname		3.23
hunspell		1.7.0
hunspell-en		0.20140811.1
hunspell-en-GB		0.20140811.1
hunspell-en-US		0.20140811.1
hunspell-filesystem		1.7.0
hwdata	0,233	0,384
info	5.1	6.7
inih		49
initscripts	9.03.58	10.09
iproute	4.4.0	6.10.0
iptables	1.4.21	
iputils	21 de dezembro de 2012	20210202
irqbalance	1.5.0	1.9.0
jansson		2.14

Pacote	AL1 AMI	AL2203 AMI
java-1.7.0-openjdk	1.7.0.321	
javapackages-tools	0.9.1	
jemalloc		5.2.1
jitterentropy		3.4.1
jpackage-utils	1.7.5	
jq		1.7.1
json-c		0,14
kbd	1.15	2.4.0
kbd-misc	1.15	2.4.0
kernel	4.14.336	6.1.112
kernel-libbpf		6.1.112
kernel-livepatch-repo-s3		2023.6.20241031
kernel-srpm-macros		1,0
kernel-tools	4.14.336	6.1.112
keyutils	1.5.8	1.6.3
keyutils-libs	1.5.8	1.6.3
kmod	14	29
kmod-libs	14	29
kpartx	0.4.9	
kpatch-runtime		0.9.7

Pacote	AL1 AMI	AL2203 AMI
krb5-libs	1.15.1	1.21.3
lcms2	2.6	
less	436	608
libacl	2.2.49	2.3.1
libaio	0.3.109	0.3.111
libarchive		3.7.4
libargon2		27 de dezembro de 2017
libassuan	2.0.3	2.5.5
libattr	2.4.46	2.5.1
libbasicobjects		0.1.1
libblkid	2.23.2	2.37.4
libcap	2,16	2,48
libcap54	2,54	
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcgroup	0,40.rc1	
libcollection		0.7.0
libcom_err	1.43.5	1.46,5
libcomps		0.1.20
libconfig		1.7.2

Pacote	AL1 AMI	AL2203 AMI
libcurl	7.61.1	
libcurl-minimal		8.5.0
libdb		5.3.28
libdhash		0.5.0
libdnf		0.69,0
libeconf		0.4.0
libedit	2.11	3.1
libev		4,33
libevent	2.0.21	2.1.12
libfdisk		2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libfontenc	1.0.5	
libgcc		11.4.1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1,42
libgssglue	0.1	
libibverbs		48,0

Pacote	AL1 AMI	AL2203 AMI
libICE	1.0.6	
libicu	50,2	
libidn	1,18	
libidn2	2.3.0	2.3.2
libini_config		1.3.1
libjpeg-turbo	1.2.90	
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libldb		2.6.2
libmaxminddb		1.5.2
libmetalink		0.1.3
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.23.2	2.37.4
libnetfilter_conntrack	1.0.4	
libnfnetlink	1.0.1	
libnfsidmap	0.25	2.5.4
libnghttp2	1.33.0	1.59.0
libnih	1.0.1	
libnl	1.1.4	

Pacote	AL1 AMI	AL2203 AMI
libnl3		3.5.0
libpath_utils		0.2.1
libpcap		1.10.1
libpipeline	1.2.3	1.5.3
libpkgconf		1.8.0
libpng	1.2.49	
libpsl	0.6.2	0,21,1
libpwquality	1.2.3	1.4.4
libref_array		0.1.5
librepo		1.14.5
libreport-filesystem		2.15.2
libseccomp		2.5.3
libselinux	2.1.10	3.4
libselinux-utils	2.1.10	3.4
libsemanage	2.1.6	3.4
libsepol	2.1.7	3.4
libsigsegv		2.13
libSM	1.2.1	
libsmartcols	2.23.2	2.37.4
libsolv		0.7.22

Pacote	AL1 AMI	AL2203 AMI
libss	1.43.5	1.46,5
libssh2	1.4.2	
libsss_certmap		2.9.4
libsss_idmap		2.9.4
libsss_nss_idmap		2.9.4
libsss_sudo		2.9.4
libstdc++		11.4.1
libstdc++72	7.2.1	
libstoragemgmt		1.9.4
libsysfs	2.1.0	
libtalloc		2.3.4
libtasn1	2.3	4.19.0
libtdb		1.4.7
libtevent		0.13.0
libtextstyle		0,21
libtirpc	0.2.4	1.3.3
libudev	173	
libunistring	0.9.3	0.9.10
libuser	0,60	0,63
libutempter	1.1.5	1.2.1

Pacote	AL1 AMI	AL2203 AMI
libuuid	2.23.2	2.37.4
libuv		1.47.0
libverto	0.2.5	0.3.2
libverto-libev		0.3.2
libX11	1.6.0	
libX11-common	1.6.0	
libXau	1.0.6	
libxcb	1.11	
libXcomposite	0.4.3	
libxcrypt		4.4.3
libXext	1.3.2	
libXfont	1.4.5	
libXi	1.7.2	
libxml2	2.9.1	2.10.4
libxml2-python27	2.9.1	
libXrender	0.9.8	
libxslt	1.1.28	
libXtst	1.2.2	
libyaml	0.1.6	0.2.5
libzstd		1.5.5

Pacote	AL1 AMI	AL2203 AMI
linux-firmware-whence		20210208
lm_sensors-libs		3.6.0
lmdb-libs		0.9.29
log4j-cve-2021-44228-hotpatch	1.3	
logrotate	3.7.8	3.20.1
lsof	4,82	4.94.0
lua	5.1.4	
lua-libs		5.4.4
lua-srpm-macros		1
lvm2	2.02.166	
lvm2-libs	2.02.166	
lz4-libs		1.9.4
mailcap	2.1.31	
make	3,82	
man-db	2.6.3	2.9.3
man-pages	4.10	5.10
mdadm	3.2.6	
microcode_ctl	2.1	2.1
mingetty	1,08	

Pacote	AL1 AMI	AL2203 AMI
mpfr		4.1.0
nano	2.5.3	5,8
nc	1,84	
ncurses	5.7	6.2
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
nettle		3.8
net-tools	1,60	2,0
newt	0.52,11	0,52,21
newt-python27	0.52,11	
nfs-utils	1.3.0	2.5.4
npth		1.6
nspr	4.25.0	4.35.0
nss	3.53.1	3.90,0
nss-pem	1.0.3	
nss-softokn	3.53.1	3.90,0
nss-softokn-freebl	3.53.1	3.90,0
nss-sysinit	3.53.1	3.90,0
nss-tools	3.53.1	
nss-util	3.53.1	3.90,0

Pacote	AL1 AMI	AL2203 AMI
ntp	4.2.8 p15	
ntpdate	4.2.8 p15	
ntsysv	1.3.49.3	1.15
numactl	2.0.7	
numactl-libs		2.0.14
ocaml-srpm-macros		6
oniguruma		6.9.7.1
openblas-srpm-macros		2
openldap	2.4.40	2.4.57
openssh	7.4p1	8,7p1
openssh-clients	7.4p1	8,7p1
openssh-server	7.4p1	8,7p1
openssl	1,0,2 k	3.0.8
openssl-libs		3.0.8
openssl-pkcs11		0.4.12
os-prober		1,7
p11-kit	0.18.5	0.24.1
p11-kit-trust	0.18.5	0.24.1
package-notes-srpm-macros		0.4
pam	1.1.8	1.5.1

Pacote	AL1 AMI	AL2203 AMI
pam_ccreds	10	
pam_krb5	2.3.11	
pam_passwdqc	1.0.5	
parted	2.1	3.4
passwd	0,79	0,80
pciutils	3.1.10	3.7.0
pciutils-libs	3.1.10	3.7.0
pcre	8.21	
pcre2		10h40
pcre2-syntax		10h40
perl	5.16.3	
perl-Carp	1,26	1,50
perl-Class-Struct		0,66
perl-constant	1,27	1,33
perl-Digest	1.17	
perl-Digest-HMAC	1,03	
perl-Digest-MD5	2,52	
perl-Digest-SHA	5,85	
perl-DynaLoader		1,47
perl-Encode	2,51	3,15

Pacote	AL1 AMI	AL2203 AMI
perl-Errno		1,30
perl-Exporter	5,68	5,74
perl-Fcntl		1.13
perl-File-Basename		2,85
perl-File-Path	2.09	2,18
perl-File-stat		1,09
perl-File-Temp	0.23.01	0.231.100
perl-Filter	1,49	
perl-Getopt-Long	2,40	2,52
perl-Getopt-Std		1.12
perl-HTTP-Tiny	0,033	0,078
perl-if		0.60.800
perl-interpreter		5.32.1
perl-IO		1,43
perl-IPC-Open3		1,21
perl-libs	5.16.3	5.32.1
perl-macros	5.16.3	
perl-MIME-Base64		3,16
perl-mro		1,23
perl-overload		1,31

Pacote	AL1 AMI	AL2203 AMI
perl-overloading		0,02
perl-parent	0,225	0,238
perl-PathTools	3,40	3,78
perl-Pod-Escapes	1.04	1,07
perl-podlators	2.5.1	4.14
perl-Pod-Perldoc	3.20	3.28.01
perl-Pod-Simple	3,28	3,42
perl-Pod-Usage	1,63	2.01
perl-POSIX		1,94
perl-Scalar-List-Utils	1,27	1,56
perl-SelectSaver		1.02
perl-Socket	2.010	2.032
perl-srpm-macros		1
perl-Storable	2,45	3.21
perl-subs		1,03
perl-Symbol		1,08
perl-Term-ANSIColor		5.01
perl-Term-Cap		1.17
perl-Text-ParseWords	3,29	3,30
perl-Text-Tabs+Wrap		2021.07.26

Pacote	AL1 AMI	AL2203 AMI
perl-threads	1,87	
perl-threads-shared	1,43	
perl-Time-HiRes	1.9725	
perl-Time-Local	1.2300	1.300
perl-vars		1,05
pinentry	0.7.6	
pkgconf		1.8.0
pkgconfig	0.27.1	
pkgconf-m4		1.8.0
pkgconf-pkg-config		1.8.0
pm-utils	1.4.1	
policycoreutils	2.1.12	3.4
policycoreutils-python-utils		3.4
popt	1.13	1,18
procmail	3.22	
procps	3.2.8	
procps-ng		3.3.17
protobuf-c		1.4.1
psacct	6.3.2	6.6.4
psmisc	22,20	23,4

Pacote	AL1 AMI	AL2203 AMI
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2.7.18	
python27-babel	0.9.4	
python27-backports	1,0	
python27-backports-ssl_match_hostname	3.4.0.2	
python27-boto	2.48.0	
python27-botocore	1.17.31	
python27-chardet	2.0.1	
python27-colorama	0.4.1	
python27-configobj	4.7.2	
python27-crypto	2.6.1	
python27-daemon	1.5.2	
python27-dateutil	2.1	
python27-devel	2.7.18	
python27-docutils	0,11	
python27-ecdsa	0,11	
python27-futures	3.0.3	
python27-imaging	1.1.6	

Pacote	AL1 AMI	AL2203 AMI
python27-iniparse	0.3.1	
python27-jinja2	2.7.2	
python27-jmespath	0.9.2	
python27-jsonpatch	1.2	
python27-jsonpointer	1,0	
python27-kitchen	1.1.1	
python27-libs	2.7.18	
python27-lockfile	0.8	
python27-markupsafe	0,11	
python27-paramiko	1.15.1	
python27-pip	9.0.3	
python27-ply	3.4	
python27-pyasn1	0.1.7	
python27-pycurl	7.19.0	
python27-pygpgme	0.3	
python27-pyliblzma	0.5.3	
python27-pystache	0.5.3	
python27-pyxattr	0.5.0	
python27-PyYAML	3.10	
python27-requests	1.2.3	

Pacote	AL1 AMI	AL2203 AMI
python27-rsa	3.4.1	
python27-setuptools	36.2.7	
python27-simplejson	3.6.5	
python27-six	1.8.0	
python27-urlgrabber	3.10	
python27-urllib3	1.24.3	
python27-virtualenv	15.1.0	
python3		3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-daemon		2.3.0
python3-dateutil		2.8.1
python3-dbus		1.2.18

Pacote	AL1 AMI	AL2203 AMI
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0,16
python3-gpg		1.15.1
python3-hawkey		0.69,0
python3-idna		(2.10)
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1,21
python3-jsonpointer		2,0
python3-jsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69,0
python3-libs		3.9.16
python3-libselinux		3.4
python3-libsemanage		3.4
python3-libstorage mgmt		1.9.4
python3-lockfile		0.12.2

Pacote	AL1 AMI	AL2203 AMI
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycoreutils		3.4
python3-prettytable		0.7.2
python3-prompt-toolkit		3.0.24
python3-pycparser		2.20
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2

Pacote	AL1 AMI	AL2203 AMI
python3-setools		4.4.1
python3-setuptools		59.6.0
python3-setuptools-wheel		59.6.0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5
python-chevron		0.13.1
python-srpm-macros		3.9
quota	4,00	4.06
quota-nls	4,00	4.06
readline	6.2	8.1
rmt	0.4	
rng-tools	5	6.14
rootfiles	8.1	8.1
rpcbind	0.2.0	1.2.6
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3

Pacote	AL1 AMI	AL2203 AMI
rpm-plugin-systemd-inhibit		4.16.1.3
rpm-python27	4.11.3	
rpm-sign-libs		4.16.1.3
rsync	3.0.6	3.2.6
rsyslog	5.8.10	
ruby	2,0	
ruby20	2.0.0.648	
ruby20-irb	2.0.0.648	
ruby20-libs	2.0.0.648	
rubygem20-bigdecimal	1.2.0	
rubygem20-json	1.8.3	
rubygem20-psych	2.0.0	
rubygem20-rdoc	4.2.2	
rubygems20	2.0.14.1	
rust-srpm-macros		21
sbsigntools		0.9.4
screen	4.0.3	4.8.0
sed	4.2.1	4.8
selinux-policy		38.1.45

Pacote	AL1 AMI	AL2203 AMI
selinux-policy-targeted		38.1.45
sendmail	8.14.4	
setserial	2,17	
setup	2.8.14	2.13.7
sgpio	1.2.0.10	
shadow-utils	4.1.4.2	4,9
shared-mime-info	1.1	
slang	2.2.1	2.3.2
sqlite	3.7.17	
sqlite-libs		3.40,0
sssd-client		2.9.4
sssd-common		2.9.4
sssd-kcm		2.9.4
sssd-nfs-idmap		2.9.4
strace		6.8
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1,0
sysfsutils	2.1.0	
sysstat		12.5.6
systemd		252,23

Pacote	AL1 AMI	AL2203 AMI
systemd-libs		252,23
systemd-networkd		252,23
systemd-pam		252,23
systemd-resolved		252,23
systemd-udev		252,23
system-release	2018.03	2023.6.20241031
systemtap-runtime		4.8
sysvinit	2,87	
tar	1,26	1,34
tbb		2020.3
tcp_wrappers	7.6	
tcp_wrappers-libs	7.6	
tcpdump		4.99.1
tcsh		24.6.07
time	1,7	1.9
tmpwatch	2.9.16	
traceroute	2.0.14	2.1.3
ttmkfdir	3.0.9	
tzdata	2023c	2024a
tzdata-java	2023c	

Pacote	AL1 AMI	AL2203 AMI
udev	173	
unzip	6.0	6.0
update-motd	1.0.1	2.2
upstart	0.6.5	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.23.2	2.37.4
util-linux-core		2.37.4
vim-common	9.0.2120	9.0.2153
vim-data	9.0.2120	9.0.2153
vim-enhanced	9.0.2120	9.0.2153
vim-filesystem	9.0.2120	9.0.2153
vim-minimal	9.0.2120	9.0.2153
wget	1,18	1.21.3
which	2,19	2.21
words	3.0	3.0
xfsdump		3.1.11
xfsprogs		5.18.0
xorg-x11-fonts-Type1	7.2	
xorg-x11-font-utils	7.2	

Pacote	AL1 AMI	AL2203 AMI
xxd	9.0.2120	9.0.2153
xxhash-libs		0.8.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
yum-plugin-upgrade-helper	1.1.31	
yum-utils	1.1.31	
zip	3.0	3.0
zlib	1.2.8	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

Comparando pacotes instalados no Amazon Linux 1 (AL1) e no Amazon Linux 2023 Minimal AMIs

Uma comparação do RPMs presente no AL1 e no AL2 023 Minimal. AMIs

Pacote	AL1 Mínimo	AL2023 Mínimo
acpid	2.0.19	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-repo-s3		2023.6.20241031
amazon-linux-sb-keys		2023.1
amd-ucode-firmware		20210208
audit	2.6.5	3.0.6
audit-libs	2.6.5	3.0.6
authconfig	6.2.8	
awscli-2		2.15.30
basesystem	10.0	11
bash	4.2.46	5.2.15
binutils	2.27	
bzip2	1.0.6	
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.62	2023.2.68
checkpolicy	2.1.10	3.4
chkconfig	1.3.49.3	
chrony		4.3

Pacote	AL1 Mínimo	AL2023 Mínimo
cloud-disk-utils	0,27	
cloud-init	0.7.6	22.2.2
cloud-init-cfg-ec2		22.2.2
cloud-utils-growpart		0,31
coreutils	8.22	8,32
coreutils-common		8,32
cpio	(2.10)	2.13
cracklib	2.8.16	2.9.6
cracklib-dicts	2.8.16	2.9.6
cronie	1.4.4	
cronie-anacron	1.4.4	
crontabs	1.10	
crypto-policies		2020428
cryptsetup-libs		2.6.1
curl	7.61.1	
curl-minimal		8.5.0
cyrus-sasl	2.1.23	
cyrus-sasl-lib	2.1.23	2.1.27
dash	0.5.5.1	
db4	4.7.25	

Pacote	AL1 Mínimo	AL2023 Mínimo
db4-utils	4.7.25	
dbus		1.12.28
dbus-broker		32
dbus-common		1.12.28
dbus-libs	1.6.12	1.12.28
device-mapper		1.02.185
device-mapper-libs		1.02.185
dhclient	4.1.1	
dhcp-common	4.1.1	
diffutils	3.3	3.8
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dracut	004	055
dracut-config-ec2		3.0
dracut-config-generic		055

Pacote	AL1 Mínimo	AL2023 Mínimo
dracut-modules-growroot	0.20	
e2fsprogs	1.43.5	1.46,5
e2fsprogs-libs	1.43.5	1.46,5
ec2-utils	0.7	2.2.0
ed	1.1	
efi-filesystem		5
efivar		38
efivar-libs		38
elfutils-default-yama-scope		0.188
elfutils-libelf	0,168	0.188
elfutils-libs		0.188
ethtool	3,15	
expat	2.1.0	2.5.0
file	5,37	5,39
file-libs	5,37	5,39
filesystem	2.4.30	3,14
findutils	4.4.2	4.8.0
fipscheck	1.3.1	
fipscheck-lib	1.3.1	

Pacote	AL1 Mínimo	AL2023 Mínimo
fuse-libs	2.9.4	2.9.9
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1,19
gdisk	0.8.10	1.0.8
generic-logos	17.0.0	
get_reference_source	1.2	
gettext		0,21
gettext-libs		0,21
glib2	2.36.3	2.74.7
glibc	2,17	2.34
glibc-all-langpacks		2.34
glibc-common	2,17	2.34
glibc-locale-source		2.34
gmp	6.0.0	6.2.1
gnupg2	2.0.28	
gnupg2-minimal		2.3.7
gnutls		3.8.0
gpgme	1.4.3	1.15.1
grep	2.20	3.8

Pacote	AL1 Mínimo	AL2023 Mínimo
groff	1.22.2	
groff-base	1.22.2	1.22.4
grub	0,97	
grub2-common		2.06
grub2-efi-x64-ec2		2.06
grub2-pc-modules		2.06
grub2-tools		2.06
grub2-tools-minimal		2.06
grubby	7.0.15	8,40
gzip	1.5	1.12
hesiod	3.1.0	
hmaccalc	0.9.12	
hostname		3.23
hwdata	0,233	0,384
info	5.1	
inih		49
initscripts	9.03.58	10.09
iproute	4.4.0	6.10.0
iptables	1.4.21	
iputils	21 de dezembro de 2012	20210202

Pacote	AL1 Mínimo	AL2023 Mínimo
irqbalance		1.9.0
jansson		2.14
jitterentropy		3.4.1
jq		1.7.1
json-c		0,14
kbd	1.15	2.4.0
kbd-misc	1.15	2.4.0
kernel	4.14.336	6.1.112
kernel-libbpf		6.1.112
kernel-livepatch-repo-s3		2023.6.20241031
keyutils-libs	1.5.8	1.6.3
kmod	14	29
kmod-libs	14	29
krb5-libs	1.15.1	1.21.3
less	436	608
libacl	2.2.49	2.3.1
libarchive		3.7.4
libargon2		27 de dezembro de 2017
libassuan	2.0.3	2.5.5
libattr	2.4.46	2.5.1

Pacote	AL1 Mínimo	AL2023 Mínimo
libblkid	2.23.2	2.37.4
libcap	2,16	2,48
libcap54	2,54	
libcap-ng	0.7.5	0.8.2
libcbor		0.7.0
libcgroup	0,40.rc1	
libcom_err	1.43.5	1.46,5
libcomps		0.1.20
libcurl	7.61.1	
libcurl-minimal		8.5.0
libdb		5.3.28
libdnf		0.69,0
libeconf		0.4.0
libedit	2.11	3.1
libfdisk		2.37.4
libffi	3.0.13	3.4.4
libfido2		1.10.0
libgcc		11.4.1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2

Pacote	AL1 Mínimo	AL2023 Mínimo
libgomp		11.4.1
libgpg-error	1.11	1,42
libicu	50,2	
libidn	1,18	
libidn2	2.3.0	2.3.2
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2.23.2	2.37.4
libnetfilter_conntrack	1.0.4	
libnfnetlink	1.0.1	
libnghttp2	1.33.0	1.59.0
libnih	1.0.1	
libpipeline		1.5.3
libpsl	0.6.2	0,21,1
libpwquality	1.2.3	1.4.4
librepo		1.14.5
libreport-filesystem		2.15.2
libseccomp		2.5.3

Pacote	AL1 Mínimo	AL2023 Mínimo
libselinux	2.1.10	3.4
libselinux-utils	2.1.10	3.4
libsemanage	2.1.6	3.4
libsepol	2.1.7	3.4
libsigsegv		2.13
libsmartcols	2.23.2	2.37.4
libsolv		0.7.22
libss	1.43.5	1.46,5
libssh2	1.4.2	
libstdc++		11.4.1
libstdc++72	7.2.1	
libsysfs	2.1.0	
libtasn1	2.3	4.19.0
libtextstyle		0,21
libudev	173	
libunistring	0.9.3	0.9.10
libuser	0,60	0,63
libutempter	1.1.5	1.2.1
libuuid	2.23.2	2.37.4
libverto	0.2.5	0.3.2

Pacote	AL1 Mínimo	AL2023 Mínimo
libxcrypt		4.4.3
libxml2	2.9.1	2.10.4
libyaml	0.1.6	0.2.5
libzstd		1.5.5
linux-firmware-whence		20210208
logrotate	3.7.8	3.20.1
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
make	3,82	
man-db		2.9.3
microcode_ctl	2.1	2.1
mingetty	1,08	
mpfr		4.1.0
ncurses	5.7	6.2
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
nettle		3.8
net-tools	1,60	2,0
newt	0.52,11	

Pacote	AL1 Mínimo	AL2023 Mínimo
newt-python27	0.52,11	
npth		1.6
nspr	4.25.0	
nss	3.53.1	
nss-pem	1.0.3	
nss-softokn	3.53.1	
nss-softokn-freebl	3.53.1	
nss-sysinit	3.53.1	
nss-tools	3.53.1	
nss-util	3.53.1	
ntp	4.2.8 p15	
ntpdate	4.2.8 p15	
numactl-libs		2.0.14
oniguruma		6.9.7.1
openldap	2.4.40	2.4.57
openssh	7.4p1	8,7p1
openssh-clients		8,7p1
openssh-server	7.4p1	8,7p1
openssl	1,0,2 k	3.0.8
openssl-libs		3.0.8

Pacote	AL1 Mínimo	AL2023 Mínimo
openssl-pkcs11		0.4.12
os-prober		1,7
p11-kit	0.18.5	0.24.1
p11-kit-trust	0.18.5	0.24.1
pam	1.1.8	1.5.1
passwd	0,79	0,80
pciutils	3.1.10	3.7.0
pciutils-libs	3.1.10	3.7.0
pcre	8.21	
pcre2		10h40
pcre2-syntax		10h40
pinentry	0.7.6	
pkgconfig	0.27.1	
policycoreutils	2.1.12	3.4
popt	1.13	1,18
procmail	3.22	
procps	3.2.8	
procps-ng		3.3.17
psmisc	22,20	23,4
pth	2.0.7	

Pacote	AL1 Mínimo	AL2023 Mínimo
publicsuffix-list-dafsa		20240212
python27	2.7.18	
python27-babel	0.9.4	
python27-backports	1,0	
python27-backports-ssl_match_hostname	3.4.0.2	
python27-chardet	2.0.1	
python27-configobj	4.7.2	
python27-iniparse	0.3.1	
python27-jinja2	2.7.2	
python27-jsonpatch	1.2	
python27-jsonpointer	1,0	
python27-libs	2.7.18	
python27-markupsafe	0,11	
python27-pycurl	7.19.0	
python27-pygpgme	0.3	
python27-pyliblzma	0.5.3	
python27-pyxattr	0.5.0	
python27-PyYAML	3.10	
python27-requests	1.2.3	

Pacote	AL1 Mínimo	AL2023 Mínimo
python27-setuptools	36.2.7	
python27-six	1.8.0	
python27-urlgrabber	3.10	
python27-urllib3	1.24.3	
python3		3.9.16
python3-attrs		20.3.0
python3-audit		3.0.6
python3-awscrt		0.19.19
python3-babel		2.9.1
python3-cffi		1.14.5
python3-chardet		4.0.0
python3-colorama		0.4.4
python3-configobj		5.0.6
python3-cryptography		36.0.1
python3-dateutil		2.8.1
python3-dbus		1.2.18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0,16

Pacote	AL1 Mínimo	AL2023 Mínimo
python3-gpg		1.15.1
python3-hawkey		0.69,0
python3-idna		(2.10)
python3-jinja2		2.11.3
python3-jmespath		0.10.0
python3-jsonpatch		1,21
python3-jsonpointer		2,0
python3-jsonschema		3.2.0
python3-libcomps		0.1.20
python3-libdnf		0.69,0
python3-libs		3.9.16
python3-libselinux		3.4
python3-libsemanage		3.4
python3-markupsafe		1.1.1
python3-netifaces		0.10.6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3.11
python3-policycoreutils		3.4
python3-prettytable		0.7.2

Pacote	AL1 Mínimo	AL2023 Mínimo
python3-prompt-toolkit		3.0.24
python3-pycparser		2.20
python3-pyrsistent		0.17.3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022.7.1
python3-pyyaml		5.4.1
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0.16.6
python3-ruamel-yaml-clib		0.1.2
python3-setools		4.4.1
python3-setuptools		59.6.0
python3-setuptools-wheel		59.6.0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1.25.10
python3-wcwidth		0.2.5

Pacote	AL1 Mínimo	AL2023 Mínimo
readline	6.2	8.1
rng-tools		6.14
rootfiles	8.1	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-plugin-selinux		4.16.1.3
rpm-plugin-systemd-inhibit		4.16.1.3
rpm-python27	4.11.3	
rpm-sign-libs		4.16.1.3
rsyslog	5.8.10	
sbsigntools		0.9.4
sed	4.2.1	4.8
selinux-policy		38.1.45
selinux-policy-targeted		38.1.45
sendmail	8.14.4	
setserial	2,17	
setup	2.8.14	2.13.7
shadow-utils	4.1.4.2	4,9

Pacote	AL1 Mínimo	AL2023 Mínimo
shared-mime-info	1.1	
slang	2.2.1	
sqlite	3.7.17	
sqlite-libs		3.40,0
sudo	1.8.23	1.9.15
sysctl-defaults	1.0	1,0
sysfsutils	2.1.0	
systemd		252,23
systemd-libs		252,23
systemd-networkd		252,23
systemd-pam		252,23
systemd-resolved		252,23
systemd-udev		252,23
system-release	2018.03	2023.6.20241031
sysvinit	2,87	
tar	1,26	1,34
tcp_wrappers-libs	7.6	
tzdata	2023c	2024a
udev	173	
update-motd	1.0.1	2.2

Pacote	AL1 Mínimo	AL2023 Mínimo
upstart	0.6.5	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2.23.2	2.37.4
util-linux-core		2.37.4
vim-data	9.0.2120	9.0.2153
vim-minimal	9.0.2120	9.0.2153
which	2,19	2.21
xfsprogs		5.18.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1.1.31	
yum-plugin-upgrade-helper	1.1.31	
zlib	1.2.8	1.2.11
zram-generator		1.1.2
zram-generator-defaults		1.1.2

Pacote	AL1 Mínimo	AL2023 Mínimo
zstd		1.5.5

Comparando pacotes instalados nas imagens de contêiner base do Amazon Linux 1 (AL1) e do Amazon Linux 2023

Uma comparação do RPMs presente nas imagens do contêiner base AL1 e AL2 023.

Pacote	AL1 Contêiner	AL2Recipiente 023
alternatives		1.15
amazon-linux-repo-cdn		2023.6.20241031
audit-libs		3.0.6
basesystem	10.0	11
bash	4.2.46	5.2.15
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023.2.62	2023.2.68
chkconfig	1.3.49.3	
coreutils	8.22	
coreutils-single		8,32
crypto-policies		2020428
curl	7.61.1	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.23	

Pacote	AL1 Contêiner	AL2Recipiente 023
db4	4.7.25	
db4-utils	4.7.25	
dnf		4.14.0
dnf-data		4.14.0
elfutils-default-yama-scope		0.188
elfutils-libelf	0,168	0.188
elfutils-libs		0.188
expat	2.1.0	2.5.0
file-libs	5,37	5,39
filesystem	2.4.30	3,14
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1,19
glib2	2.36.3	2.74.7
glibc	2,17	2.34
glibc-common	2,17	2.34
glibc-minimal-langpack		2.34
gmp	6.0.0	6.2.1
gnupg2	2.0.28	

Pacote	AL1 Contêiner	AL2Recipiente 023
gnupg2-minimal		2.3.7
gpgme	1.4.3	1.15.1
grep	2.20	3.8
gzip	1.5	
info	5.1	
json-c		0,14
keyutils-libs	1.5.8	1.6.3
krb5-libs	1.15.1	1.21.3
libacl	2.2.49	2.3.1
libarchive		3.7.4
libassuan	2.0.3	2.5.5
libattr	2.4.46	2.5.1
libblkid		2.37.4
libcap	2,16	2,48
libcap-ng		0.8.2
libcom_err	1.43.5	1.46,5
libcomps		0.1.20
libcurl	7.61.1	
libcurl-minimal		8.5.0
libdnf		0.69,0

Pacote	AL1 Contêiner	AL2Recipiente 023
libffi	3.0.13	3.4.4
libgcc		11.4.1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1,42
libicu	50,2	
libidn2	2.3.0	2.3.2
libmodulemd		2.13.0
libmount		2.37.4
libnghttp2	1.33.0	1.59.0
libpsl	0.6.2	0.21.1
librepo		1.14.5
libreport-filesystem		2.15.2
libselinux	2.1.10	3.4
libsepol	2.1.7	3.4
libsigsegv		2.13
libsmartcols		2.37.4
libsolv		0.7.22
libssh2	1.4.2	

Pacote	AL1 Contêiner	AL2Recipiente 023
libstdc++		11.4.1
libstdc++72	7.2.1	
libtasn1	2.3	4.19.0
libunistring	0.9.3	0.9.10
libuuid		2.37.4
libverto	0.2.5	0.3.2
libxcrypt		4.4.3
libxml2	2.9.1	2.10.4
libxml2-python27	2.9.1	
libyaml		0.2.5
libzstd		1.5.5
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
make	3,82	
mpfr		4.1.0
ncurses	5.7	
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
npth		1.6

Pacote	AL1 Contêiner	AL2Recipiente 023
nspr	4.25.0	
nss	3.53.1	
nss-pem	1.0.3	
nss-softokn	3.53.1	
nss-softokn-freebl	3.53.1	
nss-sysinit	3.53.1	
nss-tools	3.53.1	
nss-util	3.53.1	
openldap	2.4.40	
openssl	1,0,2 k	
openssl-libs		3.0.8
p11-kit	0.18.5	0.24.1
p11-kit-trust	0.18.5	0.24.1
pcre	8.21	
pcre2		10h40
pcre2-syntax		10h40
pinentry	0.7.6	
pkgconfig	0.27.1	
popt	1.13	1,18
pth	2.0.7	

Pacote	AL1 Contêiner	AL2Recipiente 023
publicsuffix-list-dafsa		20240212
python27	2.7.18	
python27-chardet	2.0.1	
python27-iniparse	0.3.1	
python27-kitchen	1.1.1	
python27-libs	2.7.18	
python27-pycurl	7.19.0	
python27-pygpgme	0.3	
python27-pyliblzma	0.5.3	
python27-pyxattr	0.5.0	
python27-urlgrabber	3.10	
python3		3.9.16
python3-dnf		4.14.0
python3-gpg		1.15.1
python3-hawkey		0.69,0
python3-libcomps		0.1.20
python3-libdnf		0.69,0
python3-libs		3.9.16
python3-pip-wheel		21.3.1
python3-rpm		4.16.1.3

Pacote	AL1 Contêiner	AL2Recipiente 023
python3-setuptools-wheel		59.6.0
readline	6.2	8.1
rpm	4.11.3	4.16.1.3
rpm-build-libs	4.11.3	4.16.1.3
rpm-libs	4.11.3	4.16.1.3
rpm-python27	4.11.3	
rpm-sign-libs		4.16.1.3
sed	4.2.1	4.8
setup	2.8.14	2.13.7
shared-mime-info	1.1	
sqlite	3.7.17	
sqlite-libs		3.40,0
sysctl-defaults	1,0	
system-release	2018.03	2023.6.20241031
tar	1,26	
tzdata	2023c	2024a
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-ovl	1.1.31	

Pacote	AL1 Contêiner	AL2Recipiente 023
`yum-plugin-priorities`	1.1.31	
`yum-utils`	1.1.31	
`zlib`	1.2.8	1.2.11

AL2023 requisitos do sistema

Esta seção descreve os requisitos do sistema para usar o AL2 023.

Tópicos

Requisitos de CPU para executar AL2 023

Para executar qualquer código AL2 023, o processador usado precisa atender a determinados requisitos mínimos. Tentativas de executar AL2 023 em CPUs que não atendam a esses requisitos podem resultar em erros de instrução ilegais logo no início da execução do código.

Os requisitos mínimos se aplicam a AL2023 na Amazon EC2AL2023 em containers, AL2023 fora da Amazon EC2 e.

Requisitos de CPU ARM para AL2 023

Todos os AL2 023 (`aarch64`ARM) os binários são criados para 64 bits. Não 32 bits ARM binários estão disponíveis, portanto, 64 bits ARM A CPU é necessária.

> **Note**
>
> Para instâncias baseadas em ARM, o AL2 023 suporta apenas tipos de instância que usam processadores Graviton2 ou posteriores. AL2023 não suporta instâncias A1.

AL2O 023 requer um processador compatível com ARMv8 2.2 com a extensão de criptografia (). `ARMv8.2+crypto` Todos os pacotes AL2 023 do `aarch64` são construídos com o `-march=armv8.2-a+crypto` sinalizador do compilador. Embora tentemos imprimir mensagens de erro graciosas quando o código AL2 023 tenta ser executado em versões mais antigas. ARM processadores, é possível que a primeira mensagem de erro seja um erro ilegal de instrução.

> **Note**
>
> Devido aos requisitos `aarch64` básicos de CPU AL2 023, todos Raspberry Pi sistemas anteriores ao Raspberry Pi 5 não atendem aos requisitos mínimos de CPU.

Requisitos de CPU x86-64 para 023 AL2

Todos os `x86-64` binários AL2 023 são criados para a `x86-64v2` revisão da `x86-64` arquitetura, passando `-march=x86-64-v2` para o compilador.

A `x86-64v2` revisão da arquitetura adiciona os seguintes recursos de CPU à `x86-64` arquitetura básica:

- `CMPXCHG16B`
- `LAHF-SAHF`
- `POPCNT`
- `SSE3`
- `SSE4_1`
- `SSE4_2`
- `SSSE3`

Isso é aproximadamente mapeado para `x86-64` processadores lançados em 2009 ou posteriores. Os exemplos incluem o Intel Nehalem, AMD Jaguar, Atom Silvermont, junto com o VIA Nano and Eden C microarquiteturas.

Na Amazon EC2, todos os tipos de `x86-64` instância são compatíveis`x86-64v2`, incluindo `M1C1`, e famílias de `M2` instâncias.

Não x86 de 32 bits (i686) AL2 023 binários são construídos. Embora o AL2 023 mantenha o suporte para executar binários de espaço de usuário de 32 bits, essa funcionalidade está obsoleta e pode ser removida em uma futura versão principal do Amazon Linux. Para obter mais informações, consulte Pacotes x86 (i686) de 32 bits.

Requisitos de memória (RAM) para executar o AL2 023

A EC2 `.nano` família Amazon de tipos de instância (`t2.nano`, `t3.nanot3a.nano`, `et4g.nano`) tem 512 MB de RAM, que é o requisito mínimo para AL2 023.

> **Note**
>
> Embora 512 MB seja o requisito mínimo, esses tipos de instância têm restrição de memória e a funcionalidade e o desempenho podem ser limitados.

AL2023 imagens não foram testadas em sistemas com menos de 512 MB de RAM. A execução de imagens de contêiner baseadas em AL2 023 em menos de 512 MB de RAM dependerá da carga de trabalho em contêineres.

Algumas cargas de trabalho, como `dnf upgrade` entre algumas versões AL2 023, podem exigir mais de 512 MB de RAM. Por esse motivo, a versão AL2023.3 introduziu a habilitação `zram` por padrão para instâncias com menos de 800 MB de RAM. Para cargas de trabalho em contêineres, isso significa que algumas cargas de trabalho podem funcionar bem em AL2 023 instâncias com essa quantidade de memória, mas falhar quando executadas em um contêiner restrito a essa quantidade de uso de memória.

Por exemplo, tipos com menos de 800 MB de RAM, o AL2 023 (a partir de AL2023,3 ou mais recente) habilitará a troca `zram` baseada por padrão. Exemplos de tipos de EC2 instância da Amazon com menos de 800 MB de memória incluem `t4g.nano` `t3a.nanot3.nano`,`t2.nano`,, `t1.micro` e. Isso significa menos cenários de falta de memória para esses tipos de instância, porque o AL2 023 compactará e descompactará páginas de memória sob demanda. Isso permite workloads que, de outra forma, exigiriam um tipo de instância com mais memória, às custas do uso da CPU necessário para fazer a compactação.

AL2023 Desktop gráfico

O Amazon Linux 2023 fornece uma interface gráfica opcional, leve e otimizada para a nuvem baseada no GNOME a partir da versão 2023.7. Esse ambiente de desktop moderno oferece recursos aprimorados de produtividade com ferramentas integradas, como o Firefox, para navegação segura, enquanto mantém o suporte do Amazon DCV e do VNC para acesso remoto.

Tópicos relacionados

Para obter mais informações sobre a instalação do ambiente de desktop gráfico, consulte a documentação a seguir:

- Tutorial: Instale o ambiente de trabalho GNOME em 023 AL2

Executando aplicativos em AL2 023

Esta seção aborda métodos para executar aplicativos no Amazon Linux 2023 (AL2023), incluindo o gerenciamento de quando eles são iniciados (e reiniciados) e o controle do uso de recursos.

Tópicos

Limitando o uso de recursos do processo em AL2 023 usando systemd

No Amazon Linux 2023 (AL2023), recomendamos usar `systemd` para controlar quais recursos podem ser usados por processos ou grupos de processos. `systemd`O uso é um substituto poderoso e fácil de usar para manipular `cgroups` manualmente ou usar utilitários como`cpulimit`, que antes só estavam disponíveis para o Amazon Linux em repositórios de terceirosEPEL.

Para obter informações abrangentes, consulte a `systemd` documentação inicial do systemd.resource-control ou o man página para `systemd.resource-control` em uma instância AL2 023.

Os exemplos abaixo usarão o teste de estresse da `stress-ng` CPU (do `stress-ng` pacote) para simular um aplicativo com muita CPU e `memcached` simular um aplicativo com muita memória.

Os exemplos abaixo abrangem a colocação de um limite de CPU em um comando único e um limite de memória em um serviço. A maioria das restrições de recursos `systemd` oferecidas pode ser usada em qualquer lugar `systemd` que execute um processo, e várias podem ser usadas ao mesmo tempo. Os exemplos abaixo são limitados a uma única restrição para fins ilustrativos.

Controle de recursos com **`systemd-run`** para executar comandos únicos

Embora comumente associado aos serviços do sistema, também `systemd` pode ser usado por usuários não root para executar serviços, agendar cronômetros ou executar processos pontuais. No exemplo a seguir, vamos usar `stress-ng` como nosso aplicativo de exemplo. No primeiro exemplo, vamos executá-lo usando `systemd-run` a conta `ec2-user` padrão e, no segundo exemplo, colocaremos limites no uso da CPU.

Example Use **`systemd-run`** na linha de comando para executar um processo, sem limitar o uso de recursos

1. Certifique-se de que o `stress-ng` pacote esteja instalado, pois vamos usá-lo em nosso exemplo.

   ```
   [ec2-user ~]$ sudo dnf install -y stress-ng
   ```

2. Use `systemd-run` para executar um teste de estresse da CPU de 10 segundos sem limitar a quantidade de CPU que ele pode usar.

   ```
   [ec2-user ~]$ systemd-run --user --tty --wait --property=CPUAccounting=1 stress-ng
    --cpu 1 --timeout 10
   Running as unit: run-u6.service
   Press ^] three times within 1s to disconnect TTY.
   stress-ng: info:  [339368] setting to a 10 second run per stressor
   stress-ng: info:  [339368] dispatching hogs: 1 cpu
   stress-ng: info:  [339368] successful run completed in 10.00s
   Finished with result: success
   Main processes terminated with: code=exited/status=0
   Service runtime: 10.068s
   CPU time consumed: 9.060s
   ```

   A `--user` opção diz `systemd-run` para executar o comando como o usuário com o qual estamos logados, a `--tty` opção significa um TTY está conectado, `--wait` significa esperar até que o serviço seja concluído e a `--property=CPUAccounting=1` opção instrui `systemd-run` a registrar quanto tempo de CPU é usado na execução do processo. A opção de linha de `--property` comando pode ser usada para passar `systemd-run` configurações que poderiam ser configuradas em um arquivo `systemd.unit` de configuração.

Quando instruído a colocar carga na CPU, o `stress-ng` programa usará todo o tempo de CPU disponível para realizar o teste durante o período solicitado. Para um aplicativo do mundo real, pode ser desejável limitar o tempo total de execução de um processo. No exemplo abaixo, solicitaremos que ela `stress-ng` seja executada por um período maior do que a restrição de duração máxima que impomos ao uso `systemd-run` dela.

Example Use **`systemd-run`** na linha de comando para executar um processo, limitando o uso da CPU a 1 segundo

1. Verifique se o `stress-ng` está instalado para executar este exemplo.
2. A `LimitCPU` propriedade é equivalente à `ulimit -t` qual limitará a quantidade máxima de tempo na CPU que esse processo poderá usar. Nesse caso, como estamos solicitando uma execução de estresse de 10 segundos e limitando o uso da CPU a 1 segundo, o comando receberá um `SIGXCPU` sinal e falhará.

```
[ec2-user ~]$ systemd-run --user --tty --wait --property=CPUAccounting=1 --
property=LimitCPU=1 stress-ng --cpu 1 --timeout 10
Running as unit: run-u12.service
Press ^] three times within 1s to disconnect TTY.
stress-ng: info:  [340349] setting to a 10 second run per stressor
stress-ng: info:  [340349] dispatching hogs: 1 cpu
stress-ng: fail:  [340349] cpu instance 0 corrupted bogo-ops counter, 1370 vs 0
stress-ng: fail:  [340349] cpu instance 0 hash error in bogo-ops counter and run
 flag, 3250129726 vs 0
stress-ng: fail:  [340349] metrics-check: stressor metrics corrupted, data is
 compromised
stress-ng: info:  [340349] unsuccessful run completed in 1.14s
Finished with result: exit-code
Main processes terminated with: code=exited/status=2
Service runtime: 1.201s
CPU time consumed: 1.008s
```

Mais comumente, talvez você queira restringir a porcentagem de tempo de CPU que pode ser consumida por um processo específico. No exemplo abaixo, restringiremos a porcentagem de tempo de CPU que pode ser consumida pelo`stress-ng`. Para um serviço do mundo real, pode ser desejável limitar a porcentagem máxima de tempo de CPU que um processo em segundo plano pode consumir para deixar os recursos livres para o processo que atende às solicitações do usuário.

Example Use **`systemd-run`** para limitar um processo a 10% do tempo de CPU em uma CPU

1. Verifique se o `stress-ng` está instalado para executar este exemplo.
2. Vamos usar a `CPUQuota` propriedade to tell `systemd-run` para restringir o uso da CPU para o comando que vamos executar. Não estamos limitando a quantidade de tempo que o processo pode ser executado, apenas a quantidade de CPU que ele pode usar.

```
[ec2-user ~]$ systemd-run --user --tty --wait --property=CPUAccounting=1 --
property=CPUQuota=10% stress-ng --cpu 1 --timeout 10
Running as unit: run-u13.service
Press ^] three times within 1s to disconnect TTY.
stress-ng: info:  [340664] setting to a 10 second run per stressor
stress-ng: info:  [340664] dispatching hogs: 1 cpu
stress-ng: info:  [340664] successful run completed in 10.08s
Finished with result: success
Main processes terminated with: code=exited/status=0
Service runtime: 10.140s
CPU time consumed: 1.014s
```

Observe como o CPU a contabilidade nos diz que, embora o serviço tenha sido executado por 10 segundos, ele consumiu apenas 1 segundo do tempo real de CPU.

Há várias maneiras de configurar `systemd` para limitar o uso de recursos para CPU, memória, rede e E/S. Consulte a `systemd` documentação inicial para systemd.resource-control ou o man página para `systemd.resource-control` em uma instância AL2 023 para uma documentação abrangente.

Nos bastidores, `systemd` está usando recursos do kernel Linux, como `cgroups` implementar esses limites, evitando a necessidade de configurá-los manualmente. A documentação do Linux Kernel `cgroup-v2` contém muitos detalhes sobre o `cgroups` trabalho.

Controle de recursos em um **`systemd`** serviço

Há vários parâmetros que podem ser adicionados à `[Service]` seção de `systemd` serviços para controlar o uso dos recursos do sistema. Isso inclui limites rígidos e flexíveis. Para saber o comportamento exato de cada opção, consulte a `systemd` documentação inicial do systemd.resource-control ou o man página para `systemd.resource-control` em uma instância AL2 023.

Os limites comumente usados são `MemoryHigh` especificar um limite de limitação no uso da memória, `MemoryMax` definir um limite máximo rígido (que, uma vez atingido, o OOM Killer é invocado) e `CPUQuota` (conforme ilustrado na seção anterior). Também é possível configurar pesos e prioridades em vez de números fixos.

Example Usando **`systemd`** para definir limites de uso de memória nos serviços

Neste exemplo`memcached`, definiremos um limite de uso de memória rígida para um cache simples de valores-chave e mostraremos como o OOM Killer é invocado para esse serviço e não para todo o sistema.

1. Primeiro, precisamos instalar os pacotes necessários para este exemplo.

```
[ec2-user ~]$ sudo dnf install -y memcached libmemcached-awesome-tools
```

2. Ative o `memcached.service` e, em seguida, inicie o serviço para que ele `memcached` esteja em execução.

```
[ec2-user ~]$ sudo systemctl enable memcached.service
Created symlink /etc/systemd/system/multi-user.target.wants/memcached.service # /
usr/lib/systemd/system/memcached.service.
[ec2-user ~]$ sudo systemctl start memcached.service
```

3. Verifique se `memcached.service` está em execução.

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
     Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
 disabled)
     Active: active (running) since Fri 2025-01-31 22:36:42 UTC; 1s ago
   Main PID: 356294 (memcached)
      Tasks: 10 (limit: 18907)
     Memory: 1.8M
        CPU: 20ms
     CGroup: /system.slice/memcached.service
     ##356294 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
 127.0.0.1,::1

Jan 31 22:35:36 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
 memcached.service - memcached daemon.
```

4. Agora que `memcached` está instalado e em execução, podemos observar que ele funciona inserindo alguns dados aleatórios no cache

   `/etc/sysconfig/memcached`Na `CACHESIZE` variável é definida como 64 por padrão, o que significa 64 megabytes. Ao inserir mais dados no cache do que o tamanho máximo do cache,

podemos ver que preenchemos o cache e alguns itens são removidos usando`memcached-tool`, e que estão usando cerca de 64 MB de memória. `memcached.service`

```
[ec2-user ~]$ for i in $(seq 1 150); do dd if=/dev/random of=$i bs=512k count=1;
 memcp -s localhost $i; done
[ec2-user ~]$ memcached-tool localhost display
  #  Item_Size  Max_age   Pages   Count   Full?  Evicted Evict_Time OOM
  2     120B         0s       1       0      no        0        0    0
 39   512.0K         4s      63     126     yes       24        2    0
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
     Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
 disabled)
     Active: active (running) since Fri 2025-01-31 22:36:42 UTC; 7min ago
   Main PID: 356294 (memcached)
      Tasks: 10 (limit: 18907)
     Memory: 66.7M
        CPU: 203ms
     CGroup: /system.slice/memcached.service
             ##356294 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
 127.0.0.1,::1

Jan 31 22:36:42 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
 memcached.service - memcached daemon.
```

5. Use a `MemoryMax` propriedade para definir um limite rígido para `memcached.service` onde, se atingido, o OOM Killer será invocado. Opções adicionais podem ser definidas para o serviço adicionando-as a um arquivo de substituição. Isso pode ser feito editando diretamente o `/etc/systemd/system/memcached.service.d/override.conf` arquivo ou interativamente usando o `edit` comando de`systemctl`.

```
[ec2-user ~]$ sudo systemctl edit memcached.service
```

Adicione o seguinte à substituição para definir um limite rígido de 32 MB de memória para o serviço.

```
[Service]
MemoryMax=32M
```

6. Diga `systemd` para recarregar sua configuração

```
[ec2-user ~]$ sudo systemctl daemon-reload
```

7. Observe que agora `memcached.service` está sendo executado com um limite de memória de 32 MB.

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
     Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
 disabled)
    Drop-In: /etc/systemd/system/memcached.service.d
             ##override.conf
     Active: active (running) since Fri 2025-01-31 23:09:13 UTC; 49s ago
   Main PID: 358423 (memcached)
      Tasks: 10 (limit: 18907)
     Memory: 1.8M (max: 32.0M available: 30.1M)
        CPU: 25ms
     CGroup: /system.slice/memcached.service
             ##358423 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
 127.0.0.1,::1

Jan 31 23:09:13 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
 memcached.service - memcached daemon.
```

8. O serviço funcionará normalmente usando menos de 32 MB de memória, o que podemos verificar carregando menos de 32 MB de dados aleatórios no cache e, em seguida, verificando o status do serviço.

```
[ec2-user ~]$ for i in $(seq 1 30); do dd if=/dev/random of=$i bs=512k count=1;
 memcp -s localhost $i; done
```

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
     Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
 disabled)
    Drop-In: /etc/systemd/system/memcached.service.d
             ##override.conf
     Active: active (running) since Fri 2025-01-31 23:14:48 UTC; 3s ago
   Main PID: 359492 (memcached)
      Tasks: 10 (limit: 18907)
     Memory: 18.2M (max: 32.0M available: 13.7M)
```

```
        CPU: 42ms
     CGroup: /system.slice/memcached.service
             ##359492 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
 127.0.0.1,::1

Jan 31 23:14:48 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
 memcached.service - memcached daemon.
```

9. Agora podemos `memcached` usar mais de 32 MB de memória tentando usar todos os 64 MB de cache que são a configuração padrão`memcached`.

```
[ec2-user ~]$ for i in $(seq 1 150); do dd if=/dev/random of=$i bs=512k count=1;
 memcp -s localhost $i; done
```

Você observará que em algum momento durante o comando acima, há erros de conexão com o `memcached` servidor. Isso ocorre porque o OOM Killer encerrou o processo devido à restrição que impusemos a ele. O restante do sistema funcionará normalmente e nenhum outro processo será considerado pelo OOM Killer, pois é apenas o `memcached.service` que restringimos.

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
     Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
 disabled)
    Drop-In: /etc/systemd/system/memcached.service.d
             ##override.conf
     Active: failed (Result: oom-kill) since Fri 2025-01-31 23:20:28 UTC; 2s ago
   Duration: 2.901s
    Process: 360130 ExecStart=/usr/bin/memcached -p ${PORT} -u ${USER} -m
 ${CACHESIZE} -c ${MAXCONN} $OPTIONS (code=killed, signal=KILL)
   Main PID: 360130 (code=killed, signal=KILL)
        CPU: 94ms

Jan 31 23:20:25 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
 memcached.service - memcached daemon.
Jan 31 23:20:28 ip-1-2-3-4.us-west-2.compute.internal systemd[1]:
 memcached.service: A process of this unit has been killed by the OOM killer.
Jan 31 23:20:28 ip-1-2-3-4.us-west-2.compute.internal systemd[1]:
 memcached.service: Main process exited, code=killed, status=9/KILL
Jan 31 23:20:28 ip-1-2-3-4.us-west-2.compute.internal systemd[1]:
 memcached.service: Failed with result 'oom-kill'.
```

Limitando o uso de recursos do processo em AL2 023 usando cgroups

Embora seja recomendável usarControle de recursos com systemd, esta seção aborda o uso básico dos `libcgroup-tools` utilitários básicos para limitar o uso da CPU e da memória dos processos. Ambos os métodos são alternativas ao uso do `cpulimit` utilitário, encontrado anteriormente emEPEL.

O exemplo abaixo abrange a execução do teste de `stress-ng` estresse (do `stress-ng` pacote) enquanto limita o uso de CPU e memória usando utilitários do `libcgroup-tools` pacote e os ajustáveis. `sysfs`

Use **`libcgroup-tools`** na linha de comando para limitar o uso de recursos

1. Instale o pacote `libcgroup-tools`.

   ```
   [ec2-user ~]$ sudo dnf install libcgroup-tools
   ```

2. Crie um `cgroup` com os `cpu` controladores `memory` e e dê a ele um nome (`our-example-limits`). Usando as `-t` opções `-a` e para permitir que o `ec2-user` usuário controle os ajustáveis do `cgroup`

   ```
   [ec2-user ~]$ sudo cgcreate -a ec2-user -t ec2-user -g memory,cpu:our-example-limits
   ```

   Agora existe um `/sys/fs/cgroup/our-example-limits/` diretório que contém arquivos que podem ser usados para controlar cada ajustável.

   > **Note**
   >
   > O Amazon Linux 2 usa `cgroup-v1` em vez dissocgroup-v2, o que é usado em AL2 023. Ativado AL2, os `sysfs` caminhos são diferentes `/sys/fs/cgroup/memory/our-example-limits` e haverá `/sys/fs/cgroup/cpu/our-example-limits` diretórios pertencentes aos `ec2-user` quais conterão arquivos que podem ser usados para controlar os limites docgroup.

3. Limite o uso de memória de todos os processos em nosso `cgroup` a 100 milhões de bytes.

```
[ec2-user ~]$ echo 100000000 > /sys/fs/cgroup/our-example-limits/memory.max
```

> **Note**
>
> O Amazon Linux 2 usa `cgroup-v1` em vez do `cgroup-v2` que o Amazon Linux 2023 usa. Isso significa que alguns ajustáveis são diferentes. Para limitar o uso de memória AL2, o ajustável abaixo é usado em vez disso.
>
> ```
> [ec2-user ~]$ echo 10000000 > /sys/fs/cgroup/memory/our-example-limits/
> memory.limit_in_bytes
> ```

4. Limite o uso da CPU de todos os processos em nosso `cgroup` a 10%. O formato do `cpu.max` arquivo é `$MAX $PERIOD` limitar o grupo a consumir `$MAX` cada`$PERIOD`.

   ```
   [ec2-user ~]$ echo 10000 100000 > /sys/fs/cgroup/our-example-limits/cpu.max
   ```

   O Amazon Linux 2 usa `cgroup-v1` em vez do `cgroup-v2` que o Amazon Linux 2023 usa. Isso significa que alguns ajustáveis são diferentes, incluindo como limitar o uso da CPU.

5. O exemplo abaixo é executado `stress-ng` (que pode ser instalado executando`dnf install -y stress-ng`) no `our-example-limits` cgroup. Enquanto o `stress-ng` comando está em execução, você pode observar `top` que ele está limitado a 10% de CPU hora.

   ```
   [ec2-user ~]$ sudo cgexec -g memory,cpu:our-example-limits stress-ng --cpu 1
   ```

6. Limpe removendo o cgroup

   ```
   [ec2-user ~]$ sudo cgdelete -g memory,cpu:our-example-limits
   ```

A documentação do Linux Kernel `cgroup-v2` contém muitos detalhes sobre como eles funcionam. A documentação dos controladores de CPU e memória abrange os detalhes de como usar cada opção ajustável.

Usando AL2 023 em AWS

Você pode configurar o AL2 023 para uso com outros Serviços da AWS. Por exemplo, você pode escolher uma AMI AL2 023 ao iniciar uma instância do Amazon Elastic Compute Cloud EC2 (Amazon).

Para esses procedimentos de configuração, você usa o serviço AWS Identity and Access Management (IAM). Para obter mais informações sobre o IAM, consulte os seguintes materiais de referência:

- AWS Identity and Access Management (IAM)
- Guia do usuário do IAM

Tópicos

Começando com AWS

Inscreva-se para um Conta da AWS

Se você não tiver um Conta da AWS, conclua as etapas a seguir para criar um.

Para se inscrever em um Conta da AWS

1. Abra a https://portal.aws.amazon.com/billing/inscrição.

2. Siga as instruções online.

   Parte do procedimento de inscrição envolve receber uma chamada telefônica ou uma mensagem de texto e inserir um código de verificação pelo teclado do telefone.

   Quando você se inscreve em um Conta da AWS, um Usuário raiz da conta da AWSé criado. O usuário-raiz tem acesso a todos os Serviços da AWS e recursos na conta. Como prática recomendada de segurança, atribua o acesso administrativo a um usuário e use somente o usuário-raiz para executar tarefas que exigem acesso de usuário-raiz.

AWS envia um e-mail de confirmação após a conclusão do processo de inscrição. A qualquer momento, você pode visualizar a atividade atual da sua conta e gerenciar sua conta acessando https://aws.amazon.com/e escolhendo Minha conta.

Criar um usuário com acesso administrativo

Depois de se inscrever em um Conta da AWS, proteja seu Usuário raiz da conta da AWS AWS IAM Identity Center, habilite e crie um usuário administrativo para que você não use o usuário root nas tarefas diárias.

Proteja seu Usuário raiz da conta da AWS

1. Faça login AWS Management Consolecomo proprietário da conta escolhendo Usuário raiz e inserindo seu endereço de Conta da AWS e-mail. Na próxima página, insira a senha.

   Para obter ajuda ao fazer login usando o usuário-raiz, consulte Fazer login como usuário-raiz no Guia do usuário do Início de Sessão da AWS .
2. Habilite a autenticação multifator (MFA) para o usuário-raiz.

   Para obter instruções, consulte Habilitar um dispositivo de MFA virtual para seu usuário Conta da AWS raiz (console) no Guia do usuário do IAM.

Criar um usuário com acesso administrativo

1. Habilita o Centro de Identidade do IAM.

   Para obter instruções, consulte Habilitar o AWS IAM Identity Center no Guia do usuário do AWS IAM Identity Center .
2. No Centro de Identidade do IAM, conceda o acesso administrativo a um usuário.

Para ver um tutorial sobre como usar o Diretório do Centro de Identidade do IAM como fonte de identidade, consulte Configurar o acesso do usuário com o padrão Diretório do Centro de Identidade do IAM no Guia AWS IAM Identity Center do usuário.

Iniciar sessão como o usuário com acesso administrativo

- Para fazer login com o seu usuário do Centro de Identidade do IAM, use o URL de login enviado ao seu endereço de e-mail quando o usuário do Centro de Identidade do IAM foi criado.

  Para obter ajuda para fazer login usando um usuário do IAM Identity Center, consulte Como fazer login no portal de AWS acesso no Guia Início de Sessão da AWS do usuário.

Atribuir acesso a usuários adicionais

1. No Centro de Identidade do IAM, crie um conjunto de permissões que siga as práticas recomendadas de aplicação de permissões com privilégio mínimo.

   Para obter instruções, consulte Criar um conjunto de permissões no Guia do usuário do AWS IAM Identity Center .
2. Atribua usuários a um grupo e, em seguida, atribua o acesso de autenticação única ao grupo.

   Para obter instruções, consulte Adicionar grupos no Guia do usuário do AWS IAM Identity Center .

Conceder acesso programático

Os usuários precisam de acesso programático se quiserem interagir com pessoas AWS fora do AWS Management Console. A forma de conceder acesso programático depende do tipo de usuário que está acessando AWS.

Para conceder acesso programático aos usuários, selecione uma das seguintes opções:

Qual usuário precisa de acesso programático?	Para	Por
Identidade da força de trabalho	Use credenciais temporárias para assinar solicitações	Siga as instruções da interface que deseja utilizar.

Qual usuário precisa de acesso programático?	Para	Por
(Usuários gerenciados no Centro de Identidade do IAM)	programáticas para o AWS CLI AWS SDKs, ou. AWS APIs	• Para o AWS CLI, consulte Configurando o AWS CLI para uso AWS IAM Identity Center no Guia do AWS Command Line Interface usuário. • Para AWS SDKs, ferrament as e AWS APIs, consulte a autenticação do IAM Identity Center no Guia de referênci a de ferramentas AWS SDKs e ferramentas.
IAM	Use credenciais temporári as para assinar solicitações programáticas para o AWS CLI AWS SDKs, ou. AWS APIs	Siga as instruções em Como usar credenciais temporárias com AWS recursos no Guia do usuário do IAM.

Qual usuário precisa de acesso programático?	Para	Por
IAM	(Não recomendado) Use credenciais de longo prazo para assinar solicitaç ões programáticas para o AWS CLI, AWS SDKs, ou. AWS APIs	Siga as instruções da interface que deseja utilizar. • Para isso AWS CLI, consulte Autenticação usando credenciais de usuário do IAM no Guia do AWS Command Line Interface usuário. • Para ferramentas AWS SDKs e ferramentas, consulte Autenticar usando credenciais de longo prazo no Guia de referência de ferramentas AWS SDKs e ferramentas. • Para isso AWS APIs, consulte Gerenciamento de chaves de acesso para usuários do IAM no Guia do usuário do IAM.

AL2023 na Amazon EC2

Use um dos procedimentos a seguir para iniciar uma EC2 instância da Amazon com uma AMI AL2 023. Você pode escolher a AMI padrão ou a AMI mínima. Para obter mais informações sobre as diferenças entre a AMI padrão e a AMI mínima, consulte Comparando o padrão AL2 023 (padrão) e o mínimo AMIs.

Tópicos

- Lançamento do AL2 023 usando o console da Amazon EC2
- Iniciando AL2 023 usando o parâmetro SSM e AWS CLI

- Lançamento da AMI AL2 023 mais recente usando AWS CloudFormation
- Iniciando AL2 023 usando uma ID de AMI específica
- AL2023 Depreciação e ciclo de vida da AMI
- Conexão com AL2 203 instâncias
- Comparando AL2 0,23 padrão e mínimo AMIs

Lançamento do AL2 023 usando o console da Amazon EC2

Use o EC2 console da Amazon para iniciar uma AMI AL2 023.

> **Note**
>
> Para instâncias baseadas em ARM, o AL2 023 suporta apenas tipos de instância que usam processadores Graviton2 ou posteriores. AL2023 não oferece suporte a instâncias A1.

Use as etapas a seguir para iniciar uma EC2 instância da Amazon com uma AMI AL2 023 a partir do EC2 console da Amazon.

Para executar uma EC2 instância com uma AMI AL2 023

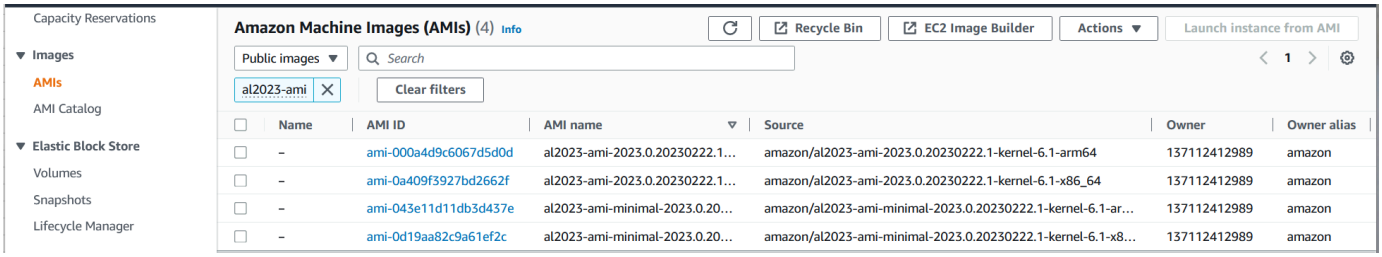
1. Abra o EC2 console da Amazon em https://console.aws.amazon.com/ec2/.
2. No painel de navegação, escolha AMIs.
3. No menu suspenso de filtros, escolha Imagens públicas.
4. No campo de pesquisa, digite **`al2023-ami`**.

   > **Note**
   >
   > Certifique-se de que a Amazon apareça na coluna Alias do proprietário.

5. Selecione uma imagem da lista. Em Origem, você pode determinar se a AMI é padrão ou mínima. Um nome de AMI AL2 023 pode ser interpretado usando este formato:

   ```
   'al2023-[ami || ami-minimal]-2023.0.[release build date].[build
   number]-kernel-[version number]-[arm64 || x86_64]'
   ```

6. A imagem a seguir mostra uma lista parcial de AL2 023 AMIs.



Para obter mais informações sobre o lançamento de EC2 instâncias da Amazon, consulte Comece a usar as instâncias do Amazon EC2 Linux no Guia EC2 do usuário da Amazon.

Iniciando AL2 023 usando o parâmetro SSM e AWS CLI

No AWS CLI, você pode usar o valor do parâmetro SSM da AMI para iniciar uma nova instância de AL2 023. Mais especificamente, use um dos valores dinâmicos do parâmetro SSM da lista a seguir e adicione `/aws/service/ami-amazon-linux-latest/` antes do valor do parâmetro SSM/. É possível usar uma instância usando a AWS CLI.

- `al2023-ami-kernel-default-arm64` para a arquitetura arm64
- `al2023-ami-minimal-kernel-default-arm64` para arquitetura arm64 (AMI mínima)
- `al2023-ami-kernel-default-x86_64` para a arquitetura x86_64
- `al2023-ami-minimal-kernel-default-x86_64` para a arquitetura x86_64 (AMI mínima)

> **Note**
>
> Cada um dos *`italic`* itens é um exemplo de parâmetro. Substitua-os por suas próprias informações.

```
$ aws ec2 run-instances \
  --image-id \
    resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86_64 \
  --instance-type m5.xlarge \
  --region us-east-1 \
  --key-name aws-key-us-east-1 \
  --security-group-ids sg-004a7650
```

O sinalizador `--image-id` especifica o valor do parâmetro SSM.

O sinalizador `--instance-type` especifica o tipo e o tamanho da instância. Esse sinalizador deve ser compatível com o tipo de AMI selecionado.

A `--region` sinalização especifica Região da AWS onde você cria sua instância.

A `--key-name` sinalização especifica Região da AWS a chave s usada para se conectar à instância. Se você não fornecer uma chave que exista na região em que você criou a instância, não poderá se conectar à instância usando SSH.

O sinalizador `--security-group-ids` especifica o grupo de segurança que determina as permissões de acesso para tráfego de rede de entrada e saída.

> **Important**
>
> Isso AWS CLI exige que você especifique um grupo de segurança existente que permita acesso à instância de sua máquina remota pela portaTCP:22. Sem um grupo de segurança especificado, sua nova instância é colocada em um grupo de segurança padrão. Em um grupo de segurança padrão, sua instância só pode se conectar às outras instâncias dentro da sua VPC.

Para obter mais informações, consulte Lançamento, listagem e encerramento de EC2 instâncias da Amazon no Guia do AWS Command Line Interface usuário.

Lançamento da AMI AL2 023 mais recente usando AWS CloudFormation

Para iniciar uma AMI AL2 023 usando AWS CloudFormation, use um dos modelos a seguir.

> **Note**
>
> `Arm64` AMIs Cada um deles exige tipos de instância diferentes. `x86_64` Para obter mais informações, consulte Tipos de EC2 instância da Amazon

Modelo `JSON`:

```
{
  "Parameters": {
```

```
    "LatestAmiId": {
      "Type": "AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>",
      "Default": "/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-
default-x86_64"
    }
  },
  "Resources": {
    "MyEC2Instance": {
      "Type": "AWS::EC2::Instance",
      "Properties": {
        "InstanceType": "t2.large",
        "ImageId": {
          "Ref": "LatestAmiId"
        }
      }
    }
  }
}
```

Modelo YAML:

```
Parameters:
  LatestAmiId:
    Type: 'AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>'
    Default: '/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-default-
x86_64'

Resources:
  Instance:
    Type: 'AWS::EC2::Instance'
    Properties:
      InstanceType: 't2.large'
      ImageId: !Ref LatestAmiId
```

Certifique-se de substituir o parâmetro AMI no final da seção “Padrão”, se necessário. Os seguintes valores de parâmetros estão disponíveis:

- `al2023-ami-kernel-6.1-arm64` para a arquitetura arm64
- `al2023-ami-minimal-kernel-6.1-arm64` para arquitetura arm64 (AMI mínima)
- `al2023-ami-kernel-6.1-x86_64` para a arquitetura x86_64
- `al2023-ami-minimal-kernel-6.1-x86_64` para a arquitetura x86_64 (AMI mínima)

A seguir estão as especificações dinâmicas do kernel. A versão padrão do kernel muda automaticamente com cada atualização da versão principal do kernel.

- `al2023-ami-kernel-default-arm64` para a arquitetura arm64
- `al2023-ami-minimal-kernel-default-arm64` para arquitetura arm64 (AMI mínima)
- `al2023-ami-kernel-default-x86_64` para a arquitetura x86_64
- `al2023-ami-minimal-kernel-default-x86_64` para a arquitetura x86_64 (AMI mínima)

Iniciando AL2 023 usando uma ID de AMI específica

Você pode iniciar uma AMI AL2 023 específica usando a ID da AMI. Você pode determinar qual ID de AMI AL2 023 é necessária consultando a lista de AMI no EC2 console da Amazon. Ou, você pode usar AWS Systems Manager. Se você estiver usando o Systems Manager, certifique-se de selecionar o alias da AMI dentre os listados na seção anterior. Para obter mais informações, consulte Consultar a AMI mais recente do Amazon Linux IDs usando o AWS Systems Manager Parameter Store.

AL2023 Depreciação e ciclo de vida da AMI

Cada nova versão AL2 023 inclui uma nova AMI. Quando a AMI é registrada, ela é marcada com uma data de suspensão de uso. A data de suspensão de uso de cada AL2 AMI 023 é de 90 dias a partir do momento em que foi lançada, de acordo com o período oferecido para cada lançamento Atualização do Kernel Live em 023 AL2 individual do kernel.

> **Note**
>
> A data de suspensão de uso de 90 dias se refere a uma AMI individual e não ao período de suporte de AL2 023 Cadência de lançamento ou do produto.

Para obter mais informações sobre a suspensão de uso da AMI, consulte Descontinuar uma AMI no Guia do usuário da Amazon. EC2

O uso regular de uma AMI atualizada para executar uma instância garante que a instância comece com as atualizações de segurança mais recentes, incluindo um kernel atualizado. Se você iniciar uma versão anterior de uma AMI e aplicar atualizações, haverá um período em que a instância não terá as atualizações de segurança mais recentes. Para garantir que você esteja usando a AMI mais recente, recomendamos usar os parâmetros do SSM.

Para obter mais informações sobre como usar os parâmetros do SSM para iniciar uma instância, consulte:

- Iniciando AL2 023 usando o parâmetro SSM e AWS CLI
- Lançamento da AMI AL2 023 mais recente usando AWS CloudFormation

Conexão com AL2 203 instâncias

Use SSH ou AWS Systems Manager para se conectar à sua instância AL2 023.

Conectar a sua instância usando SSH

Para obter instruções sobre como usar o SSH para se conectar a uma instância, consulte Conecte-se à sua instância Linux usando SSH no Guia EC2 do usuário da Amazon.

Conecte-se à sua instância usando AWS Systems Manager

Para obter instruções sobre como usar AWS Systems Manager para se conectar a uma instância AL2 023, consulte Conecte-se à sua instância Linux usando o Session Manager no Guia do EC2 usuário da Amazon.

Usando o Amazon EC2 Instance Connect

A AMI AL2 023, excluindo a AMI mínima, vem com o agente Instance EC2 Connect instalado por padrão. Para usar o EC2 Instance Connect com uma instância AL2 023 executada a partir da AMI mínima, você deve instalar o `ec2-instance-connect` pacote. Para obter instruções sobre como usar o EC2 Instance Connect, consulte Conecte-se à sua instância Linux com o EC2 Instance Connect no Guia EC2 do usuário da Amazon.

Comparando AL2 0,23 padrão e mínimo AMIs

Você pode iniciar uma EC2 instância da Amazon com uma AMI padrão (padrão) ou mínima de AL2 023. Para obter instruções sobre como iniciar uma EC2 instância da Amazon com o tipo de AMI padrão ou mínimo, consulteAL2023 na Amazon EC2.

A AMI AL2 023 padrão vem com todos os aplicativos e ferramentas mais usados instalados. Recomendamos a AMI padrão se você quiser começar rapidamente e não estiver interessado em personalizar a AMI.

A AMI AL2 023 mínima é a versão básica e simplificada que contém somente as ferramentas e utilitários mais básicos necessários para executar o sistema operacional (SO). Recomendamos a

AMI mínima se você quiser ter o menor espaço de sistema operacional possível. A AMI mínima oferece uma utilização ligeiramente reduzida do espaço em disco e melhor eficiência de custos a longo prazo. A AMI mínima é adequada se você deseja um sistema operacional menor e não se importa em instalar ferramentas e aplicativos manualmente.

A imagem do contêiner está mais próxima da AMI mínima de AL2 023 no conjunto de pacotes.

Comparar pacotes instalados em imagens Amazon Linux 2023

Uma comparação do RPMs presente nas imagens AL2 023 AMI, Minimal AMI e Container.

Pacote	AMI	AMI mínima	Contêiner
`acl`	2.3.1		
`acpid`	2.0.32		
`alternatives`	1.15	1.15	1.15
`amazon-chrony-config`	4.3	4.3	
`amazon-ec2-net-utils`	2.5.1	2.5.1	
`amazon-linux-repo-cdn`			2023.6.20241031
`amazon-linux-repo-s3`	2023.6.20241031	2023.6.20241031	
`amazon-linux-sb-keys`	2023.1	2023.1	
`amazon-rpm-config`	228		
`amazon-ssm-agent`	3.3.987.0		

Pacote	AMI	AMI mínima	Contêiner
amd-ucode-firmware	20210208 (noarca)	20210208 (noarca)	
at	3.1.23		
attr	2.5.1		
audit	3.0.6	3.0.6	
audit-libs	3.0.6	3.0.6	3.0.6
aws-cfn-bootstrap	2,0		
awscli-2	2.15.30	2.15.30	
basesystem	11	11	11
bash	5.2.15	5.2.15	5.2.15
bash-completion	2.11		
bc	1.07.1		
bind-libs	9.18.28		
bind-license	9.18.28		
bind-utils	9.18.28		
binutils	2,39		
boost-filesystem	1.75.0		
boost-system	1.75.0		
boost-thread	1.75.0		

Pacote	AMI	AMI mínima	Contêiner
bzip2	1.0.8		
bzip2-libs	1.0.8	1.0.8	1.0.8
ca-certificates	2023.2.68	2023.2.68	2023.2.68
c-ares	1.19.1		
checkpolicy	3.4	3.4	
chkconfig	1.15		
chrony	4.3	4.3	
cloud-init	22.2.2	22.2.2	
cloud-init-cfg-ec2	22.2.2	22.2.2	
cloud-utils-growpart	0,31	0,31	
coreutils	8,32	8,32	
coreutils-common	8,32	8,32	
coreutils-single			8,32
cpio	2.13	2.13	
cracklib	2.9.6	2.9.6	
cracklib-dicts	2.9.6	2.9.6	
crontabs	1.11		
crypto-policies	2020428	2020428	2020428

Pacote	AMI	AMI mínima	Contêiner
crypto-policies-scripts	2020428		
cryptsetup	2.6.1		
cryptsetup-libs	2.6.1	2.6.1	
curl-minimal	8.5.0	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27	
cyrus-sasl-plain	2.1.27		
dbus	1.12.28	1.12.28	
dbus-broker	32	32	
dbus-common	1.12.28	1.12.28	
dbus-libs	1.12.28	1.12.28	
device-mapper	1.02.185	1.02.185	
device-mapper-libs	1.02.185	1.02.185	
diffutils	3.8	3.8	
dnf	4.14.0	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2	
dnf-plugins-core	4.3.0	4.3.0	

Pacote	AMI	AMI mínima	Contêiner
dnf-plugin-support-info	1.2	1.2	
dnf-utils	4.3.0		
dosfstools	4.2		
dracut	055	055	
dracut-config-ec2	3.0	3.0	
dracut-config-generic	055	055	
dwz	0,14		
dyninst	10.2.1		
e2fsprogs	1.46,5	1.46,5	
e2fsprogs-libs	1.46,5	1.46,5	
ec2-hibinit-agent	1.0.8		
ec2-instance-connect	1.1		
ec2-instance-connect-selinux	1.1		
ec2-utils	2.2.0	2.2.0	
ed	1.14.2		
efi-filesystem	5	5	

Pacote	AMI	AMI mínima	Contêiner
efi-srpm-macros	5		
efivar	38	38	
efivar-libs	38	38	
elfutils-debuginfod-client	0.188		
elfutils-default-yama-scope	0.188	0.188	0.188
elfutils-libelf	0.188	0.188	0.188
elfutils-libs	0.188	0.188	0.188
ethtool	5.15		
expat	2.5.0	2.5.0	2.5.0
file	5,39	5,39	
file-libs	5,39	5,39	5,39
filesystem	3,14	3,14	3,14
findutils	4.8.0	4.8.0	
fonts-srpm-macros	2.0.5		
fstrm	0.6.1		
fuse-libs	2.9.9	2.9.9	
gawk	5.1.0	5.1.0	5.1.0

Pacote	AMI	AMI mínima	Contêiner
gdbm-libs	1,19	1,19	1,19
gdisk	1.0.8	1.0.8	
gettext	0,21	0,21	
gettext-libs	0,21	0,21	
ghc-srpm-macros	1.5.0		
glib2	2.74.7	2.74.7	2.74.7
glibc	2.34	2.34	2.34
glibc-all-langpacks	2.34	2.34	
glibc-common	2.34	2.34	2.34
glibc-gconv-extra	2.34		
glibc-locale-source	2.34	2.34	
glibc-minimal-langpack			2.34
gmp	6.2.1	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7	2.3.7
gnutls	3.8.0	3.8.0	
go-srpm-macros	3.2.0		
gpgme	1.15.1	1.15.1	1.15.1
gpm-libs	1.20.7		

Pacote	AMI	AMI mínima	Contêiner
grep	3.8	3.8	3.8
groff-base	1.22.4	1.22.4	
grub2-common	2.06	2.06	
grub2-efi-aa64-ec2	2.06 (64 de março)	2.06 (64 de março)	
grub2-efi-x64-ec2	2,06 (x86_64)	2,06 (x86_64)	
grub2-pc-modules	2.06	2.06	
grub2-tools	2.06	2.06	
grub2-tools-minimal	2.06	2.06	
grubby	8,40	8,40	
gssproxy	0.8.4		
gzip	1.12	1.12	
hostname	3.23	3.23	
hunspell	1.7.0		
hunspell-en	0.20140811.1		
hunspell-en-GB	0.20140811.1		
hunspell-en-US	0.20140811.1		
hunspell-filesystem	1.7.0		

Pacote	AMI	AMI mínima	Contêiner
hwdata	0,384	0,384	
info	6.7		
inih	49	49	
initscripts	10.09	10.09	
iproute	6.10.0	6.10.0	
iputils	20210202	20210202	
irqbalance	1.9.0	1.9.0	
jansson	2.14	2.14	
jemalloc	5.2.1		
jitterentropy	3.4.1	3.4.1	
jq	1.7.1	1.7.1	
json-c	0,14	0,14	0,14
kbd	2.4.0	2.4.0	
kbd-misc	2.4.0	2.4.0	
kernel	6.1.112	6.1.112	
kernel-libbpf	6.1.112	6.1.112	
kernel-livepatch-repo-s3	2023.6.20241031	2023.6.20241031	
kernel-srpm-macros	1,0		

Pacote	AMI	AMI mínima	Contêiner
kernel-tools	6.1.112		
keyutils	1.6.3		
keyutils-libs	1.6.3	1.6.3	1.6.3
kmod	29	29	
kmod-libs	29	29	
kpatch-runtime	0.9.7		
krb5-libs	1.21.3	1.21.3	1.21.3
less	608	608	
libacl	2.3.1	2.3.1	2.3.1
libaio	0.3.111		
libarchive	3.7.4	3.7.4	3.7.4
libargon2	27 de dezembro de 2017	27 de dezembro de 2017	
libassuan	2.5.5	2.5.5	2.5.5
libattr	2.5.1	2.5.1	2.5.1
libbasicobjects	0.1.1		
libblkid	2.37.4	2.37.4	2.37.4
libcap	2,48	2,48	2,48
libcap-ng	0.8.2	0.8.2	0.8.2
libcbor	0.7.0	0.7.0	
libcollection	0.7.0		

Pacote	AMI	AMI mínima	Contêiner
libcom_err	1.46,5	1.46,5	1.46,5
libcomps	0.1.20	0.1.20	0.1.20
libconfig	1.7.2		
libcurl-minimal	8.5.0	8.5.0	8.5.0
libdb	5.3.28	5.3.28	
libdhash	0.5.0		
libdnf	0.69,0	0.69,0	0.69,0
libeconf	0.4.0	0.4.0	
libedit	3.1	3.1	
libev	4,33		
libevent	2.1.12		
libfdisk	2.37.4	2.37.4	
libffi	3.4.4	3.4.4	3.4.4
libfido2	1.10.0	1.10.0	
libgcc	11.4.1	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2	1.10.2
libgomp	11.4.1	11.4.1	11.4.1
libgpg-error	1,42	1,42	1,42
libibverbs	48,0		
libidn2	2.3.2	2.3.2	2.3.2

Pacote	AMI	AMI mínima	Contêiner
libini_config	1.3.1		
libkcapi	1.4.0	1.4.0	
libkcapi-hmaccalc	1.4.0	1.4.0	
libldb	2.6.2		
libmaxminddb	1.5.2		
libmetalink	0.1.3		
libmnl	1.0.4	1.0.4	
libmodulemd	2.13.0	2.13.0	2.13.0
libmount	2.37.4	2.37.4	2.37.4
libnfsidmap	2.5.4		
libnghttp2	1.59.0	1.59.0	1.59.0
libnl3	3.5.0		
libpath_utils	0.2.1		
libpcap	1.10.1		
libpipeline	1.5.3	1.5.3	
libpkgconf	1.8.0		
libpsl	0.21.1	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4	
libref_array	0.1.5		
librepo	1.14.5	1.14.5	1.14.5

Pacote	AMI	AMI mínima	Contêiner
libreport-filesystem	2.15.2	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3	
libselinux	3.4	3.4	3.4
libselinux-utils	3.4	3.4	
libsemanage	3.4	3.4	
libsepol	3.4	3.4	3.4
libsigsegv	2.13	2.13	2.13
libsmartcols	2.37.4	2.37.4	2.37.4
libsolv	0.7.22	0.7.22	0.7.22
libss	1.46,5	1.46,5	
libsss_certmap	2.9.4		
libsss_idmap	2.9.4		
libsss_nss_idmap	2.9.4		
libsss_sudo	2.9.4		
libstdc++	11.4.1	11.4.1	11.4.1
libstoragemgmt	1.9.4		
libtalloc	2.3.4		
libtasn1	4.19.0	4.19.0	4.19.0

Pacote	AMI	AMI mínima	Contêiner
libtdb	1.4.7		
libtevent	0.13.0		
libtextstyle	0,21	0,21	
libtirpc	1.3.3		
libunistring	0.9.10	0.9.10	0.9.10
libuser	0,63	0,63	
libutempter	1.2.1	1.2.1	
libuuid	2.37.4	2.37.4	2.37.4
libuv	1.47.0		
libverto	0.3.2	0.3.2	0.3.2
libverto-libev	0.3.2		
libxcrypt	4.4.3	4.4.3	4.4.3
libxml2	2.10.4	2.10.4	2.10.4
libyaml	0.2.5	0.2.5	0.2.5
libzstd	1.5.5	1.5.5	1.5.5
linux-firmware-whence	20210208 (noarca)	20210208 (noarca)	
lm_sensors-libs	3.6.0		
lmdb-libs	0.9.29		
logrotate	3.20.1	3.20.1	
lsof	4.94.0		

Pacote	AMI	AMI mínima	Contêiner
lua-libs	5.4.4	5.4.4	5.4.4
lua-srpm-macros	1		
lz4-libs	1.9.4	1.9.4	1.9.4
man-db	2.9.3	2.9.3	
man-pages	5.10		
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)	
mpfr	4.1.0	4.1.0	4.1.0
nano	5,8		
ncurses	6.2	6.2	
ncurses-base	6.2	6.2	6.2
ncurses-libs	6.2	6.2	6.2
nettle	3.8	3.8	
net-tools	2,0	2.0	
newt	0,52,21		
nfs-utils	2.5.4		
npth	1,6	1,6	1.6
nspr	4.35.0		
nss	3.90,0		
nss-softokn	3.90,0		
nss-softokn-freebl	3.90,0		

Pacote	AMI	AMI mínima	Contêiner
nss-sysinit	3.90,0		
nss-util	3.90,0		
ntsysv	1.15		
numactl-libs	2.0.14	2.0.14	
ocaml-srpm-macros	6		
oniguruma	6.9.7.1	6.9.7.1	
openblas-srpm-macros	2		
openldap	2.4.57	2.4.57	
openssh	8,7p1	8,7p1	
openssh-clients	8,7p1	8,7p1	
openssh-server	8,7p1	8,7p1	
openssl	3.0.8	3.0.8	
openssl-libs	3.0.8	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12	
os-prober	1,7	1,7	
p11-kit	0.24.1	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1	0.24.1
package-notes-srpm-macros	0.4		

Pacote	AMI	AMI mínima	Contêiner
pam	1.5.1	1.5.1	
parted	3.4		
passwd	0,80	0,80	
pciutils	3.7.0	3.7.0	
pciutils-libs	3.7.0	3.7.0	
pcre2	10h40	10h40	10h40
pcre2-syntax	10h40	10h40	10h40
perl-Carp	1,50		
perl-Class-Struct	0,66		
perl-constant	1,33		
perl-DynaLoader	1,47		
perl-Encode	3,15		
perl-Errno	1,30		
perl-Exporter	5,74		
perl-Fcntl	1.13		
perl-File-Basename	2,85		
perl-File-Path	2,18		
perl-File-stat	1,09		
perl-File-Temp	0.231.100		

Pacote	AMI	AMI mínima	Contêiner
perl-Getopt-Long	2,52		
perl-Getopt-Std	1.12		
perl-HTTP-Tiny	0,078		
perl-if	0.60.800		
perl-interpreter	5.32.1		
perl-IO	1,43		
perl-IPC-Open3	1,21		
perl-libs	5.32.1		
perl-MIME-Base64	3,16		
perl-mro	1,23		
perl-overload	1,31		
perl-overloading	0,02		
perl-parent	0,238		
perl-PathTools	3,78		
perl-Pod-Escapes	1,07		
perl-podlators	4.14		
perl-Pod-Perldoc	3.28.01		

Pacote	AMI	AMI mínima	Contêiner
perl-Pod-Simple	3,42		
perl-Pod-Usage	2.01		
perl-POSIX	1,94		
perl-Scalar-List-Utils	1,56		
perl-SelectSaver	1.02		
perl-Socket	2.032		
perl-srpm-macros	1		
perl-Storable	3.21		
perl-subs	1,03		
perl-Symbol	1,08		
perl-Term-ANSIColor	5.01		
perl-Term-Cap	1.17		
perl-Text-ParseWords	3,30		
perl-Text-Tabs+Wrap	2021.07.26		
perl-Time-Local	1.300		
perl-vars	1,05		
pkgconf	1.8.0		

Pacote	AMI	AMI mínima	Contêiner
pkgconf-m4	1.8.0		
pkgconf-pkg-config	1.8.0		
policycoreutils	3.4	3.4	
policycoreutils-python-utils	3.4		
popt	1,18	1,18	1,18
procps-ng	3.3.17	3.3.17	
protobuf-c	1.4.1		
psacct	6.6.4		
psmisc	23,4	23,4	
publicsuffix-list-dafsa	2024/02/12	2024/02/12	2024/02/12
python3	3.9.16	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0	
python3-audit	3.0.6	3.0.6	
python3-awscrt	0.19.19	0.19.19	
python3-babel	2.9.1	2.9.1	
python3-cffi	1.14.5	1.14.5	
python3-chardet	4.0.0	4.0.0	

Pacote	AMI	AMI mínima	Contêiner
python3-colorama	0.4.4	0.4.4	
python3-configobj	5.0.6	5.0.6	
python3-cryptography	36.0.1	36.0.1	
python3-daemon	2.3.0		
python3-dateutil	2.8.1	2.8.1	
python3-dbus	1.2.18	1.2.18	
python3-distro	1.5.0	1.5.0	
python3-dnf	4.14.0	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0	
python3-docutils	0,16	0,16	
python3-gpg	1.15.1	1.15.1	1.15.1
python3-hawkey	0.69,0	0.69,0	0.69,0
python3-idna	(2.10)	(2.10)	
python3-jinja2	2.11.3	2.11.3	
python3-jmespath	0.10.0	0.10.0	

Pacote	AMI	AMI mínima	Contêiner
python3-jsonpatch	1,21	1,21	
python3-jsonpointer	2,0	2.0	
python3-jsonschema	3.2.0	3.2.0	
python3-libcomps	0.1.20	0.1.20	0.1.20
python3-libdnf	0.69,0	0.69,0	0.69,0
python3-libs	3.9.16	3.9.16	3.9.16
python3-libselinux	3.4	3.4	
python3-libsemanage	3.4	3.4	
python3-libstoragemgmt	1.9.4		
python3-lockfile	0.12.2		
python3-markupsafe	1.1.1	1.1.1	
python3-netifaces	0.10.6	0.10.6	
python3-oauthlib	3.0.2	3.0.2	

Pacote	AMI	AMI mínima	Contêiner
python3-pip-wheel	21.3.1	21.3.1	21.3.1
python3-ply	3.11	3.11	
python3-policycoreutils	3.4	3.4	
python3-prettytable	0.7.2	0.7.2	
python3-prompt-toolkit	3.0.24	3.0.24	
python3-pycparser	2.20	2.20	
python3-pyrsistent	0.17.3	0.17.3	
python3-pyserial	3.4	3.4	
python3-pysocks	1.7.1	1.7.1	
python3-pytz	2022.7.1	2022.7.1	
python3-pyyaml	5.4.1	5.4.1	
python3-requests	2.25.1	2.25.1	
python3-rpm	4.16.1.3	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6	

Pacote	AMI	AMI mínima	Contêiner
python3-ruamel-yaml-clib	0.1.2	0.1.2	
python3-setools	4.4.1	4.4.1	
python3-setuptools	59.6.0	59.6.0	
python3-setuptools-wheel	59.6.0	59.6.0	59.6.0
python3-six	1.15.0	1.15.0	
python3-systemd	235	235	
python3-urllib3	1.25.10	1.25.10	
python3-wcwidth	0.2.5	0.2.5	
python-chevron	0.13.1		
python-srpm-macros	3.9		
quota	4.06		
quota-nls	4.06		
readline	8.1	8.1	8.1
rng-tools	6.14	6.14	
rootfiles	8.1	8.1	
rpcbind	1.2.6		
rpm	4.16.1.3	4.16.1.3	4.16.1.3

Pacote	AMI	AMI mínima	Contêiner
rpm-build-libs	4.16.1.3	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3	
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3	
rpm-sign-libs	4.16.1.3	4.16.1.3	4.16.1.3
rsync	3.2.6		
rust-srpm-macros	21		
sbsigntools	0.9.4	0.9.4	
screen	4.8.0		
sed	4.8	4.8	4.8
selinux-policy	38.1.45	38.1.45	
selinux-policy-targeted	38.1.45	38.1.45	
setup	2.13.7	2.13.7	2.13.7
shadow-utils	4,9	4,9	
slang	2.3.2		
sqlite-libs	3.40,0	3.40,0	3.40,0
sssd-client	2.9.4		
sssd-common	2.9.4		

Pacote	AMI	AMI mínima	Contêiner
sssd-kcm	2.9.4		
sssd-nfs-idmap	2.9.4		
strace	6.8		
sudo	1.9.15	1.9.15	
sysctl-defaults	1.0	1,0	
sysstat	12.5.6		
systemd	252,23	252,23	
systemd-libs	252,23	252,23	
systemd-networkd	252,23	252,23	
systemd-pam	252,23	252,23	
systemd-resolved	252,23	252,23	
systemd-udev	252,23	252,23	
system-release	2023.6.20241031	2023.6.20241031	2023.6.20241031
systemtap-runtime	4.8		
tar	1,34	1,34	
tbb	2020.3		
tcpdump	4.99.1		
tcsh	24.6.07		

Pacote	AMI	AMI mínima	Contêiner
time	1.9		
traceroute	2.1.3		
tzdata	2024a	2024a	2024a
unzip	6.0		
update-motd	2.2	2.2	
userspace-rcu	0.12.1	0.12.1	
util-linux	2.37.4	2.37.4	
util-linux-core	2.37.4	2.37.4	
vim-common	9.0.2153		
vim-data	9.0.2153	9.0.2153	
vim-enhanced	9.0.2153		
vim-filesystem	9.0.2153		
vim-minimal	9.0.2153	9.0.2153	
wget	1.21.3		
which	2.21	2.21	
words	3.0		
xfsdump	3.1.11		
xfsprogs	5.18.0	5.18.0	
xxd	9.0.2153		
xxhash-libs	0.8.0		

Pacote	AMI	AMI mínima	Contêiner
xz	5.2.5	5.2.5	
xz-libs	5.2.5	5.2.5	5.2.5
yum	4.14.0	4.14.0	4.14.0
zip	3.0		
zlib	1.2.11	1.2.11	1.2.11
zram-generator	1.1.2	1.1.2	
zram-generator-defaults	1.1.2	1.1.2	
zstd	1.5.5	1.5.5	

Usando AL2 023 em contêineres

> **Note**
>
> Para obter mais informações sobre como usar o AL2 023 para hospedar cargas de trabalho em contêineres no Amazon ECS, consulte. AL2023 para hosts de contêineres do Amazon ECS

Existem várias maneiras pelas quais o AL2 023 pode ser usado dentro de contêineres, dependendo do caso de uso. AL2Imagem do contêiner base 023É mais semelhante a uma imagem de contêiner Amazon Linux 2 e à AMI mínima de AL2 023.

Para usuários avançados, oferecemos uma imagem mínima de contêiner, introduzida na versão AL2 023.2, junto com a documentação que descreve como criar contêineres básicos.

AL2O 023 também pode ser usado para hospedar cargas de trabalho em contêineres, seja de imagens de contêiner baseadas em AL2 023 ou contêineres baseados em outras distribuições Linux. Você pode usar AL2023 para hosts de contêineres do Amazon ECS ou usar diretamente os pacotes

de tempo de execução do contêiner fornecidos. Os `nerdctl` pacotes `dockercontainerd`,, e estão disponíveis para serem instalados e usados em AL2 023.

Tópicos

Usando a imagem do contêiner base AL2 023

A imagem do contêiner AL2 023 é criada a partir dos mesmos componentes de software incluídos na AL2 AMI 023. Está disponível para uso em qualquer ambiente como imagem base para workloads do Docker. Se você estiver usando o Amazon Linux AMI para aplicativos no Amazon Elastic Compute Cloud (Amazon EC2), você pode conteinerizar seus aplicativos com a imagem de contêiner Amazon Linux.

Use a imagem do contêiner Amazon Linux em seu ambiente de desenvolvimento local e, em seguida, envie seu aplicativo para AWS usar o Amazon Elastic Container Service (Amazon ECS). Para obter mais informações, consulte Usar imagens do Amazon ECR com o Amazon ECS no Guia do usuário do Amazon Elastic Container Registry.

A imagem de contêiner do Amazon Linux está disponível no Amazon ECR Public. Você pode fornecer feedback sobre AL2 023 por meio de seu AWS representante designado ou registrando um problema no repositório amazon-linux-2023 em. GitHub

Para extrair a imagem de contêiner do Amazon Linux do Amazon ECR Public

1. Autentique o cliente do Docker para seu registro do Amazon Linux Public. Os tokens de autenticação são válidos por 12 horas. Para obter mais informações, consulte Autenticação de registro privado no Guia do Usuário do Registro de Contêineres da Amazon Elastic.

> **Note**
>
> O get-login-password comando é suportado usando a versão mais recente da AWS CLI versão 2. Para obter mais informações, consulte Instalar a AWS Command Line Interface no Guia do usuário da AWS Command Line Interface .

```
$ aws ecr-public get-login-password --region us-east-1 | docker login --username
 AWS --password-stdin public.ecr.aws
```

A saída é a seguinte:

```
Login succeeded
```

2. Extraia a imagem do contêiner do Amazon Linux usando o comando docker pull. Para visualizar a imagem do contêiner do Amazon Linux na Galeria Pública do Amazon ECR, consulte Galeria pública do Amazon ECR - amazonlinux.

> **Note**
>
> Ao extrair a imagem do Docker contêiner AL2 023, você pode usar as tags em um dos seguintes formatos:
>
> - Para obter a versão mais recente da imagem do contêiner AL2 023, use a `:2023` tag.
> - Para obter uma versão específica do AL2 023, você pode usar o seguinte formato:
>   - `:2023.`*`[0-7 release quarter]`*`.`*`[release date]`*`.`*`[build number]`*
>
> Os exemplos a seguir usam a tag `:2023` e extraem a imagem de contêiner mais recente disponível de AL2 023.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023
```

3. (Opcional) Execute o contêiner localmente.

```
$ docker run -it --security-opt seccomp=unconfined public.ecr.aws/amazonlinux/
amazonlinux:2023 /bin/bash
```

Para extrair a imagem do contêiner AL2 023 do Hub Docker

1. Extraia a imagem do contêiner AL2 023 usando o docker pull comando.

   ```
   $ docker pull amazonlinux:2023
   ```

2. (Opcional) Execute o contêiner localmente.

   ```
   $ docker run -it amazonlinux:2023 /bin/bash
   ```

   > **Note**
   >
   > A imagem do contêiner de AL2 023 usa somente o gerenciador de `dnf` pacotes para instalar pacotes de software. Isso significa que não há nenhum comando `amazon-linux-extras` ou um comando equivalente a ser usado para software adicional.

AL2023 Imagem mínima do contêiner

> **Note**
>
> As imagens de contêiner AL2 023 padrão são adequadas para a maioria dos casos de uso, e a adaptação à imagem mínima do contêiner provavelmente será mais trabalhosa do que a adaptação à imagem de contêiner base AL2 023.

A imagem de contêiner mínimo AL2 023, introduzida em AL2 023.2, difere da imagem de contêiner base porque contém somente os pacotes mínimos necessários para instalar outros pacotes. A imagem mínima do contêiner foi projetada para ser um conjunto mínimo de pacotes, não um conjunto conveniente de pacotes.

A imagem mínima do contêiner AL2 023 é criada a partir de componentes de software já disponíveis em AL2 023. A principal diferença na imagem mínima do contêiner é usá-la `microdnf` para fornecer o gerenciador de `dnf` pacotes, em vez de uma imagem totalmente Python baseada em recursos`dnf`.

Isso permite que a imagem mínima do contêiner seja menor com a desvantagem de não ter o conjunto completo de recursos do gerenciador de `dnf` pacotes que está incluído na imagem AL2 023 AMIs e na imagem do contêiner base.

A imagem mínima do contêiner AL2 023 forma a base do ambiente de execução do `provided.al2023` AWS Lambda.

Para obter uma lista detalhada dos pacotes incluídos na imagem mínima do contêiner, consulte Comparar pacotes instalados em imagens de contêiner do Amazon Linux 2023.

Tamanho mínimo da imagem do contêiner

Como a imagem do contêiner mínimo AL2 023 contém menos pacotes do que a imagem do contêiner base AL2 023, ela também é significativamente menor. A tabela a seguir compara as opções de imagem de contêiner das versões atuais e anteriores do Amazon Linux.

> **Note**
>
> O tamanho da imagem é mostrado no Amazon Linux na Galeria Pública do Amazon ECR.

Imagem	Versão	Tamanho da imagem	Observação
Amazon Linux (1AL1)	2018.03.0.20230918.0	62,3 MB	Somente `x86-64`
Amazon Linux 2	2.0.20230926.0	64,2 MB	`aarch64` é 1,6 MB maior que `x86-64`
Imagem de contêiner base do Amazon Linux 2023	2023.2.20231002.0	52,4 MB	
Imagem de contêiner mínimo do Amazon Linux	2023.2.20231002.0-minimal	35,2 MB	

Usando a imagem AL2 023 Minimal Container

A imagem de contêiner mínimo AL2 023 está disponível em ECR e a `2023-minimal` tag sempre apontará para a imagem de contêiner mínimo baseada em AL2 023 mais recente, enquanto a `minimal` tag pode ser atualizada para uma versão mais recente do Amazon Linux que a 023. AL2

Você pode extrair essas tags usando `docker` o exemplo a seguir:

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:minimal
```

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
```

O exemplo a seguir mostra uma `Dockerfile` que pega a imagem mínima do contêiner e instala o GCC em cima dela:

```
FROM public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
RUN dnf install -y gcc && dnf clean all
```

Criando imagens básicas de contêineres 023 AL2

A imagem do contêiner AL2 023 é criada a partir dos mesmos componentes de software incluídos na AL2 AMI 023. Ele inclui um software que permite que a camada básica do contêiner se comporte de forma semelhante à execução em uma EC2 instância da Amazon, como o gerenciador de pacotes`dnf`. Esta seção explica como você pode construir um contêiner do zero que inclua somente as dependências mínimas necessárias para um aplicativo.

> **Note**
>
> As imagens padrão do contêiner AL2 023 são adequadas para a maioria dos casos de uso. O uso da imagem de contêiner padrão facilita a criação em cima da imagem. Uma imagem de contêiner básica dificulta a criação sobre sua imagem.

Para criar um contêiner com dependências mínimas para um aplicativo

1. Determine suas dependências de tempo de execução. Isso variará de acordo com sua inscrição.

2. Construa um `Dockerfile` / `Containerfile` que constrói `FROM scratch`. O exemplo a seguir de `Dockerfile` pode ser usado para criar um contêiner que contém somente `bash` shell e suas dependências.

```
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
RUN mkdir /sysroot
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
  --installroot /sysroot \
  -y \
  --setopt=install_weak_deps=False \
  install bash

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/bin/bash"]
```

   - Este `Dockerfile` funciona ao:

     1. Iniciando um contêiner AL2 023 chamado`build`. Esse contêiner será usado para inicializar o contêiner básico. Esse contêiner não é implantado sozinho, mas gera o contêiner a ser implantado.
     2. Criar o diretório `/sysroot`. Esse diretório será onde o contêiner `build` instalará as dependências necessárias para o contêiner de barebones. Em uma etapa subsequente, o `/sysroot` caminho será empacotado para ser o diretório raiz de nossa imagem básica.

        Usar a `--installroot` opção dessa `dnf` maneira é como criamos as outras AL2 023 imagens. Trata-se de um recurso de `dnf` que permite que instaladores e ferramentas de criação de imagens funcionem.
     3. Invocar `dnf` para instalar pacotes em `/sysroot`.

        O comando `rpm -q system-release --qf '%{VERSION}'` consulta (`-q`) o pacote `%{VERSION}`, definindo o formato da consulta (`--qf`) para imprimir a versão do pacote que está sendo consultado (a variável `system-release` é a variável `rpm` da versão do RPM).

Ao definir o argumento `--releasever` de `dnf` para a versão de `system-release` no contêiner `build`, o `Dockerfile` pode ser usado para reconstruir o contêiner básico sempre que uma imagem base de contêiner atualizada do Amazon Linux for lançada.

É possível definir o `--releasever` para qualquer versão do Amazon Linux 2023, como 2023.7.20250331. Isso significaria que o `build` contêiner seria executado como a versão AL2 023 mais recente, mas construiria o contêiner básico a partir de 2023.7.20250331, independentemente de qual fosse a versão 023 atual. AL2

A opção `--setopt=install_weak_deps=False` de configuração diz `dnf` para instalar somente as dependências necessárias, em vez de recomendadas ou sugeridas.

4. Copiar o sistema instalado na raiz de um contêiner vazio (`FROM scratch`).
5. Definindo `ENTRYPOINT` o como o binário desejado, neste caso `/bin/bash`.

3. Crie um diretório vazio e adicione o conteúdo do exemplo na Etapa 2 a um arquivo chamado `Dockerfile`.

```
$ mkdir al2023-barebones-bash-example
 $ cd al2023-barebones-bash-example
 $ cat > Dockerfile <<EOF
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
RUN mkdir /sysroot
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
  --installroot /sysroot \
  -y \
  --setopt=install_weak_deps=False \
  install bash && dnf --installroot /sysroot clean all

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/bin/bash"]
EOF
```

4. Crie o contêiner executando o comando a seguir.

```
$ docker build -t al2023-barebones-bash-example
```

5. Execute o contêiner usando o comando a seguir para ver o quão mínimo é um contêiner de somente `bash`.

```
$ docker run -it --rm al2023-barebones-bash-example
bash-5.2# rpm
bash: rpm: command not found
bash-5.2# du -sh /usr/
bash: du: command not found
bash-5.2# ls
bash: ls: command not found
bash-5.2# echo /bin/*
/bin/alias /bin/bash /bin/bashbug /bin/bashbug-64 /bin/bg /bin/catchsegv /bin/cd /
bin/command /bin/fc /bin/fg /bin/gencat /bin/getconf /bin/getent /bin/getopts /
bin/hash /bin/iconv /bin/jobs /bin/ld.so /bin/ldd /bin/locale /bin/localedef /
bin/pldd /bin/read /bin/sh /bin/sotruss /bin/sprof /bin/type /bin/tzselect /bin/
ulimit /bin/umask /bin/unalias /bin/wait /bin/zdump
```

Para um exemplo mais prático, o procedimento a seguir cria um contêiner para um aplicativo C que exibe `Hello World!`.

1. Crie um diretório vazio e adicione o código-fonte C e `Dockerfile`.

```
$ mkdir al2023-barebones-c-hello-world-example
$ cd al2023-barebones-c-hello-world-example
$ cat > hello-world.c <<EOF
#include <stdio.h>
int main(void)
{
  printf("Hello World!\n");
  return 0;
}
EOF

$ cat > Dockerfile <<EOF
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
COPY hello-world.c /
RUN dnf -y install gcc
RUN gcc -o hello-world hello-world.c
RUN mkdir /sysroot
RUN mv hello-world /sysroot/
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
```

```
  --installroot /sysroot \
  -y \
  --setopt=install_weak_deps=False \
  install glibc && dnf --installroot /sysroot clean all

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/hello-world"]
EOF
```

2. Execute o contêiner usando o seguinte comando.

```
$ docker build -t al2023-barebones-c-hello-world-example .
```

3. Execute o contêiner usando o seguinte comando.

```
$ docker run -it --rm al2023-barebones-c-hello-world-example
Hello World!
```

Comparar pacotes instalados em imagens de contêiner do Amazon Linux 2023

Uma comparação do RPMs presente na imagem do contêiner base AL2 023 em comparação com o RPMs presente na imagem do contêiner mínimo AL2 023.

Pacote	Contêiner	Contêiner mínimo
`alternatives`	1.15	1.15
`amazon-linux-repo-cdn`	2023.6.20241031	2023.6.20241031
`audit-libs`	3.0.6	3.0.6
`basesystem`	11	11
`bash`	5.2.15	5.2.15

Pacote	Contêiner	Contêiner mínimo
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.68	2023.2.68
coreutils-single	8,32	8,32
crypto-policies	2020428	2020428
curl-minimal	8.5.0	8.5.0
dnf	4.14.0	
dnf-data	4.14.0	4.14.0
elfutils-default-yama-scope	0.188	
elfutils-libelf	0.188	
elfutils-libs	0.188	
expat	2.5.0	
file-libs	5,39	5,39
filesystem	3,14	3,14
gawk	5.1.0	5.1.0
gdbm-libs	1,19	
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-common	2.34	2.34
glibc-minimal-langpack	2.34	2.34

Pacote	Contêiner	Contêiner mínimo
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gobject-introspection		1.73.0
gpgme	1.15.1	1.15.1
grep	3.8	3.8
json-c	0,14	0,14
keyutils-libs	1.6.3	1.6.3
krb5-libs	1.21.3	1.21.3
libacl	2.3.1	2.3.1
libarchive	3.7.4	3.7.4
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libblkid	2.37.4	2.37.4
libcap	2,48	2,48
libcap-ng	0.8.2	0.8.2
libcom_err	1.46,5	1.46,5
libcomps	0.1.20	
libcurl-minimal	8.5.0	8.5.0
libdnf	0.69,0	0.69,0
libffi	3.4.4	3.4.4

Pacote	Contêiner	Contêiner mínimo
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	
libgpg-error	1,42	1,42
libidn2	2.3.2	2.3.2
libmodulemd	2.13.0	2.13.0
libmount	2.37.4	2.37.4
libnghttp2	1.59.0	1.59.0
libpeas		1.32.0
libpsl	0,21,1	0,21,1
librepo	1.14.5	1.14.5
libreport-filesystem	2.15.2	2.15.2
libselinux	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22
libstdc++	11.4.1	11.4.1
libtasn1	4.19.0	4.19.0
libunistring	0.9.10	0.9.10

Pacote	Contêiner	Contêiner mínimo
libuuid	2.37.4	2.37.4
libverto	0.3.2	0.3.2
libxcrypt	4.4.3	
libxml2	2.10.4	2.10.4
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
lua-libs	5.4.4	5.4.4
lz4-libs	1.9.4	1.9.4
microdnf		3.10.0
microdnf-dnf		3.10.0
mpfr	4.1.0	4.1.0
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
npth	1,6	1.6
openssl-libs	3.0.8	3.0.8
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
pcre2	10h40	10h40
pcre2-syntax	10h40	10h40
popt	1,18	1,18

Pacote	Contêiner	Contêiner mínimo
publicsuffix-list-dafsa	2024/02/12	2024/02/12
python3	3.9.16	
python3-dnf	4.14.0	
python3-gpg	1.15.1	
python3-hawkey	0.69,0	
python3-libcomps	0.1.20	
python3-libdnf	0.69,0	
python3-libs	3.9.16	
python3-pip-wheel	21.3.1	
python3-rpm	4.16.1.3	
python3-setuptools-wheel	59.6.0	
readline	8.1	8.1
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	
rpm-libs	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	
sed	4.8	4.8
setup	2.13.7	2.13.7
sqlite-libs	3.40,0	3.40,0

Pacote	Contêiner	Contêiner mínimo
system-release	2023.6.20241031	2023.6.20241031
tzdata	2024a	
xz-libs	5.2.5	5.2.5
yum	4.14.0	
zlib	1.2.11	1.2.11

Comparar pacotes instalados em imagens de contêiner do Amazon Linux 2023

Uma comparação do RPMs presente na AMI AL2 023 Minimial com o RPMs presente na base AL2 023 e nas imagens mínimas do contêiner.

Pacote	AMI mínima	Contêiner	Contêiner mínimo
alternatives	1.15	1.15	1.15
amazon-chrony-config	4.3		
amazon-ec2-net-utils	2.5.1		
amazon-linux-repo-cdn		2023.6.20241031	2023.6.20241031
amazon-linux-repo-s3	2023.6.20241031		
amazon-linux-sb-keys	2023.1		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
amd-ucode-firmware	20210208 (noarca)		
audit	3.0.6		
audit-libs	3.0.6	3.0.6	3.0.6
awscli-2	2.15.30		
basesystem	11	11	11
bash	5.2.15	5.2.15	5.2.15
bzip2-libs	1.0.8	1.0.8	1.0.8
ca-certificates	2023.2.68	2023.2.68	2023.2.68
checkpolicy	3.4		
chrony	4.3		
cloud-init	22.2.2		
cloud-init-cfg-ec2	22.2.2		
cloud-utils-growpart	0,31		
coreutils	8,32		
coreutils-common	8,32		
coreutils-single		8,32	8,32
cpio	2.13		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
cracklib	2.9.6		
cracklib-dicts	2.9.6		
crypto-policies	2020428	2020428	2020428
cryptsetup-libs	2.6.1		
curl-minimal	8.5.0	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27		
dbus	1.12.28		
dbus-broker	32		
dbus-common	1.12.28		
dbus-libs	1.12.28		
device-mapper	1.02.185		
device-mapper-libs	1.02.185		
diffutils	3.8		
dnf	4.14.0	4.14.0	
dnf-data	4.14.0	4.14.0	4.14.0
dnf-plugin-release-notification	1.2		
dnf-plugins-core	4.3.0		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
dnf-plugin-support-info	1.2		
dracut	055		
dracut-config-ec2	3.0		
dracut-config-generic	055		
e2fsprogs	1.46,5		
e2fsprogs-libs	1.46,5		
ec2-utils	2.2.0		
efi-filesystem	5		
efivar	38		
efivar-libs	38		
elfutils-default-yama-scope	0.188	0.188	
elfutils-libelf	0.188	0.188	
elfutils-libs	0.188	0.188	
expat	2.5.0	2.5.0	
file	5,39		
file-libs	5,39	5,39	5,39
filesystem	3,14	3,14	3,14

Pacote	AMI mínima	Contêiner	Contêiner mínimo
findutils	4.8.0		
fuse-libs	2.9.9		
gawk	5.1.0	5.1.0	5.1.0
gdbm-libs	1,19	1,19	
gdisk	1.0.8		
gettext	0,21		
gettext-libs	0,21		
glib2	2.74.7	2.74.7	2.74.7
glibc	2.34	2.34	2.34
glibc-all-langpacks	2.34		
glibc-common	2.34	2.34	2.34
glibc-locale-source	2.34		
glibc-minimal-langpack		2.34	2.34
gmp	6.2.1	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7	2.3.7
gnutls	3.8.0		
gobject-introspection			1.73.0
gpgme	1.15.1	1.15.1	1.15.1

Pacote	AMI mínima	Contêiner	Contêiner mínimo
grep	3.8	3.8	3.8
groff-base	1.22.4		
grub2-common	2.06		
grub2-efi-aa64-ec2	2.06 (64 de março)		
grub2-efi-x64-ec2	2,06 (x86_64)		
grub2-pc-modules	2.06		
grub2-tools	2.06		
grub2-tools-minimal	2.06		
grubby	8,40		
gzip	1.12		
hostname	3.23		
hwdata	0,384		
inih	49		
initscripts	10.09		
iproute	6.10.0		
iputils	20210202		
irqbalance	1.9.0		
jansson	2.14		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
jitterentropy	3.4.1		
jq	1.7.1		
json-c	0,14	0,14	0,14
kbd	2.4.0		
kbd-misc	2.4.0		
kernel	6.1.112		
kernel-libbpf	6.1.112		
kernel-livepatch-repo-s3	2023.6.20241031		
keyutils-libs	1.6.3	1.6.3	1.6.3
kmod	29		
kmod-libs	29		
krb5-libs	1.21.3	1.21.3	1.21.3
less	608		
libacl	2.3.1	2.3.1	2.3.1
libarchive	3.7.4	3.7.4	3.7.4
libargon2	27 de dezembro de 2017		
libassuan	2.5.5	2.5.5	2.5.5
libattr	2.5.1	2.5.1	2.5.1

Pacote	AMI mínima	Contêiner	Contêiner mínimo
libblkid	2.37.4	2.37.4	2.37.4
libcap	2,48	2,48	2,48
libcap-ng	0.8.2	0.8.2	0.8.2
libcbor	0.7.0		
libcom_err	1.46,5	1.46,5	1.46,5
libcomps	0.1.20	0.1.20	
libcurl-minimal	8.5.0	8.5.0	8.5.0
libdb	5.3.28		
libdnf	0.69,0	0.69,0	0.69,0
libeconf	0.4.0		
libedit	3.1		
libfdisk	2.37.4		
libffi	3.4.4	3.4.4	3.4.4
libfido2	1.10.0		
libgcc	11.4.1	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2	1.10.2
libgomp	11.4.1	11.4.1	
libgpg-error	1,42	1,42	1,42
libidn2	2.3.2	2.3.2	2.3.2
libkcapi	1.4.0		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
libkcapi-hmaccalc	1.4.0		
libmnl	1.0.4		
libmodulemd	2.13.0	2.13.0	2.13.0
libmount	2.37.4	2.37.4	2.37.4
libnghttp2	1.59.0	1.59.0	1.59.0
libpeas			1.32.0
libpipeline	1.5.3		
libpsl	0,21,1	0,21,1	0,21,1
libpwquality	1.4.4		
librepo	1.14.5	1.14.5	1.14.5
libreport-filesystem	2.15.2	2.15.2	2.15.2
libseccomp	2.5.3		
libselinux	3.4	3.4	3.4
libselinux-utils	3.4		
libsemanage	3.4		
libsepol	3.4	3.4	3.4
libsigsegv	2.13	2.13	2.13
libsmartcols	2.37.4	2.37.4	2.37.4

Pacote	AMI mínima	Contêiner	Contêiner mínimo
libsolv	0.7.22	0.7.22	0.7.22
libss	1.46,5		
libstdc++	11.4.1	11.4.1	11.4.1
libtasn1	4.19.0	4.19.0	4.19.0
libtextstyle	0,21		
libunistring	0.9.10	0.9.10	0.9.10
libuser	0,63		
libutempter	1.2.1		
libuuid	2.37.4	2.37.4	2.37.4
libverto	0.3.2	0.3.2	0.3.2
libxcrypt	4.4.3	4.4.3	
libxml2	2.10.4	2.10.4	2.10.4
libyaml	0.2.5	0.2.5	0.2.5
libzstd	1.5.5	1.5.5	1.5.5
linux-firmware-whence	20210208 (noarca)		
logrotate	3.20.1		
lua-libs	5.4.4	5.4.4	5.4.4
lz4-libs	1.9.4	1.9.4	1.9.4
man-db	2.9.3		
microcode_ctl	2.1 (x86_64)		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
microdnf			3.10.0
microdnf-dnf			3.10.0
mpfr	4.1.0	4.1.0	4.1.0
ncurses	6.2		
ncurses-base	6.2	6.2	6.2
ncurses-libs	6.2	6.2	6.2
nettle	3.8		
net-tools	2,0		
npth	1,6	1,6	1.6
numactl-libs	2.0.14		
oniguruma	6.9.7.1		
openldap	2.4.57		
openssh	8,7p1		
openssh-clients	8,7p1		
openssh-server	8,7p1		
openssl	3.0.8		
openssl-libs	3.0.8	3.0.8	3.0.8
openssl-pkcs11	0.4.12		
os-prober	1,7		
p11-kit	0.24.1	0.24.1	0.24.1

Pacote	AMI mínima	Contêiner	Contêiner mínimo
p11-kit-trust	0.24.1	0.24.1	0.24.1
pam	1.5.1		
passwd	0,80		
pciutils	3.7.0		
pciutils-libs	3.7.0		
pcre2	10h40	10h40	10h40
pcre2-syntax	10h40	10h40	10h40
policycoreutils	3.4		
popt	1,18	1,18	1,18
procps-ng	3.3.17		
psmisc	23,4		
publicsuffix-list-dafsa	2024/02/12	2024/02/12	2024/02/12
python3	3.9.16	3.9.16	
python3-attrs	20.3.0		
python3-audit	3.0.6		
python3-awscrt	0.19.19		
python3-babel	2.9.1		
python3-cffi	1.14.5		
python3-chardet	4.0.0		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
python3-colorama	0.4.4		
python3-configobj	5.0.6		
python3-cryptography	36.0.1		
python3-dateutil	2.8.1		
python3-dbus	1.2.18		
python3-distro	1.5.0		
python3-dnf	4.14.0	4.14.0	
python3-dnf-plugins-core	4.3.0		
python3-docutils	0,16		
python3-gpg	1.15.1	1.15.1	
python3-hawkey	0.69,0	0.69,0	
python3-idna	(2.10)		
python3-jinja2	2.11.3		
python3-jmespath	0.10.0		
python3-jsonpatch	1,21		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
python3-jsonpointer	2,0		
python3-jsonschema	3.2.0		
python3-libcomps	0.1.20	0.1.20	
python3-libdnf	0.69,0	0.69,0	
python3-libs	3.9.16	3.9.16	
python3-libselinux	3.4		
python3-libsemanage	3.4		
python3-markupsafe	1.1.1		
python3-netifaces	0.10.6		
python3-oauthlib	3.0.2		
python3-pip-wheel	21.3.1	21.3.1	
python3-ply	3.11		
python3-policycoreutils	3.4		
python3-prettytable	0.7.2		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
python3-prompt-toolkit	3.0.24		
python3-pycparser	2.20		
python3-pyrsistent	0.17.3		
python3-pyserial	3.4		
python3-pysocks	1.7.1		
python3-pytz	2022.7.1		
python3-pyyaml	5.4.1		
python3-requests	2.25.1		
python3-rpm	4.16.1.3	4.16.1.3	
python3-ruamel-yaml	0.16.6		
python3-ruamel-yaml-clib	0.1.2		
python3-setools	4.4.1		
python3-setuptools	59.6.0		
python3-setuptools-wheel	59.6.0	59.6.0	

Pacote	AMI mínima	Contêiner	Contêiner mínimo
python3-six	1.15.0		
python3-systemd	235		
python3-urllib3	1.25.10		
python3-wcwidth	0.2.5		
readline	8.1	8.1	8.1
rng-tools	6.14		
rootfiles	8.1		
rpm	4.16.1.3	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3	
rpm-libs	4.16.1.3	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3		
rpm-plugin-systemd-inhibit	4.16.1.3		
rpm-sign-libs	4.16.1.3	4.16.1.3	
sbsigntools	0.9.4		
sed	4.8	4.8	4.8
selinux-policy	38.1.45		
selinux-policy-targeted	38.1.45		
setup	2.13.7	2.13.7	2.13.7

Pacote	AMI mínima	Contêiner	Contêiner mínimo
shadow-utils	4,9		
sqlite-libs	3.40,0	3.40,0	3.40,0
sudo	1.9.15		
sysctl-defaults	1,0		
systemd	252,23		
systemd-libs	252,23		
systemd-networkd	252,23		
systemd-pam	252,23		
systemd-resolved	252,23		
systemd-udev	252,23		
system-release	2023.6.20241031	2023.6.20241031	2023.6.20241031
tar	1,34		
tzdata	2024a	2024a	
update-motd	2.2		
userspace-rcu	0.12.1		
util-linux	2.37.4		
util-linux-core	2.37.4		
vim-data	9.0.2153		
vim-minimal	9.0.2153		

Pacote	AMI mínima	Contêiner	Contêiner mínimo
`which`	2.21		
`xfsprogs`	5.18.0		
`xz`	5.2.5		
`xz-libs`	5.2.5	5.2.5	5.2.5
`yum`	4.14.0	4.14.0	
`zlib`	1.2.11	1.2.11	1.2.11
`zram-generator`	1.1.2		
`zram-generator-defaults`	1.1.2		
`zstd`	1.5.5		

AL2023 não AWS Elastic Beanstalk

AWS Elastic Beanstalk é um serviço para implantar e escalar aplicativos e serviços da web. Carregue seu código e o Elastic Beanstalk lida automaticamente na implantação, desde o provisionamento de capacidade, balanceamento de carga e ajuste de escala automático ao monitoramento da saúde do aplicativo. Para mais informações, consulte AWS Elastic Beanstalk.

Para usar o Elastic Beanstalk, crie uma aplicação, faça upload de uma versão dela na forma de um pacote de origem (por exemplo, arquivo Java .war) no Elastic Beanstalk e forneça algumas informações sobre a aplicação. O Elastic Beanstalk inicia automaticamente um ambiente e cria e AWS configura os recursos necessários para executar seu código. Para obter mais informações, consulte o Guia do desenvolvedor do AWS Elastic Beanstalk.

As plataformas Linux do Elastic Beanstalk EC2 usam instâncias da Amazon, e essas instâncias executam o Amazon Linux. A partir de 4 de agosto de 2023, o Elastic Beanstalk oferece as seguintes ramificações de plataforma com base no Amazon Linux 2023:Docker, Tomcat, Java SE, Node.js, PHP e Python. O Elastic Beanstalk está trabalhando para liberar o suporte para AL2 0.23 em mais plataformas do Elastic Beanstalk.

https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/WelcomeA lista completa do suporte à plataforma Elastic Beanstalk e das plataformas atuais criadas com base em 023 pode ser encontrada na seção de plataformas AL2 Linux do Elastic Beanstalk do Guia do Desenvolvedor do Elastic Beanstalk.

Você pode encontrar as notas de lançamento das novas plataformas do Elastic Beanstalk e versões das plataformas existentes nas Notas de lançamento do Elastic Beanstalk.

Usando AL2 0,23 em AWS CloudShell

AWS CloudShell é um shell pré-autenticado baseado em navegador que você pode iniciar diretamente do. AWS Management Console Você pode navegar CloudShell de AWS Management Console algumas maneiras diferentes. Para obter mais informações, consulte Como começar a usar AWS CloudShell?

AWS CloudShell, que atualmente é baseado no Amazon Linux 2, migrará para o AL2 023. A migração para AL2 023 começará a ser implementada ao todo a Regiões da AWS partir de 4 de dezembro de 2023. Para obter mais informações sobre a CloudShell migração para AL2 023, consulte AWS CloudShell Migração do Amazon Linux 2 para o Amazon Linux 2023.

Usando o Amazon ECS baseado em AL2 023 AMIs para hospedar cargas de trabalho em contêineres

> **Note**
>
> Para obter mais informações sobre como usar AL2 023 dentro de um contêiner, consulteAL2023 em containers.

O Amazon Elastic Container Service (Amazon ECS) é um serviço totalmente gerenciado de orquestração de contêineres ajuda a implantar, gerenciar e dimensionar facilmente aplicações conteinerizadas. Como um serviço totalmente gerenciado, o Amazon ECS vem com as melhores práticas operacionais e de AWS configuração incorporadas. Ele é integrado a ferramentas tanto AWS quanto a de terceiros, como o Amazon Elastic Container Registry (Amazon ECR) e o Docker. Essa integração torna mais fácil para as equipes se concentrarem na criação das aplicações, não no ambiente. Você pode executar e dimensionar suas workloads de contêiner em Regiões AWS na nuvem, sem a complexidade de gerenciar um ambiente de gerenciamento ou nós.

Você pode hospedar cargas de trabalho em contêineres em AL2 023 usando a AMI otimizada para AL2 Amazon ECS baseada em 023. Para obter mais informações, consulte a AMI otimizada para Amazon ECS

Alterações em AL2 2023 para o Amazon ECS em comparação com AL2

Da mesma forma AL2, o AL2 023 fornece os pacotes básicos necessários para execução como uma instância Linux do Amazon ECS. No AL2 `containerd`,`docker`, e `ecs-init` os pacotes estavam disponíveis por meio de`amazon-linux-extras`, enquanto o AL2 023 inclui esses pacotes nos repositórios principais.

Com o recurso de atualizações determinísticas por meio de repositórios versionados, cada 023 AL2 AMI, por padrão, está bloqueada para uma versão específica do repositório. Isso também vale para a AMI AL2 otimizada 023 do Amazon ECS. Todas as atualizações do seu ambiente podem ser cuidadosamente gerenciadas e testadas antes da implantação, além de fornecer uma maneira fácil de voltar ao conteúdo de uma AMI anterior no caso de um problema. Para obter mais informações sobre esse recurso AL2 023, consulteAtualizações determinísticas por meio de repositórios versionados em 023 AL2.

AL2023 muda para cgroup v2 pela interface cgroup v1 suportada em. AL2 Para obter mais informações, consulte Hierarquia unificada do Grupo de Controle (cgroup v2).

> **Note**
>
> AL2As versões 023 anteriores à 2023.2.20230920 (a primeira versão AL2 023.2) continham um bug na manipulação de `systemd` for Out-of-Memory (OOM) dentro de um cgroup. Todos os processos no cgroup sempre foram eliminados, em vez de o OOM-killer escolher um processo por vez, que é o comportamento pretendido.
> Essa foi uma regressão quando comparada ao AL2 comportamento e foi corrigida na versão 2023.2.20230920 de 0.23. AL2

O código para criar a AMI otimizada para Amazon ECS está disponível no amazon-ecs-ami GitHub projeto. As notas de lançamento descrevem qual versão AL2 023 é mapeada para qual versão do Amazon ECS AMI.

Personalização da AMI otimizada para Amazon AL2 ECS baseada em 023

> **Important**
>
> Recomendamos que você use a AL2 AMI 023 otimizada do Amazon ECS. Para obter mais informações, consulte a AMI otimizada para Amazon ECS no Guia do desenvolvedor do Amazon Elastic Container Service.

Você pode usar os mesmos scripts de construção que o Amazon ECS usa para criar itens personalizados AMIs. Para obter mais informações, consulte o script de construção da AMI Linux otimizado para Amazon ECS.

Usando o Amazon Elastic File System em AL2 023

O Amazon Elastic File System (Amazon EFS) fornece armazenamento de arquivos sem servidor e com elasticidade total para você compartilhar dados de arquivos sem provisionar ou gerenciar a capacidade e o desempenho do armazenamento. O Amazon EFS foi criado para escalar sob demanda para petabytes sem interromper os aplicativos, podendo aumentar ou diminuir à medida que arquivos são adicionados e removidos. O Amazon EFS possui uma interface de serviços da web que permite criar e configurar sistemas de arquivos de maneira rápida e fácil. O serviço gerencia toda a infraestrutura de armazenamento de arquivos para você, para que você evite a complexidade de implantar, corrigir e manter configurações complexas de sistemas de arquivos.

O Amazon EFS oferece suporte ao protocolo Network File System versão 4 (NFSv4.1 e NFSv4 .0), portanto, os aplicativos e ferramentas que você usa atualmente funcionam perfeitamente com o Amazon EFS. Várias instâncias computacionais, incluindo Amazon EC2, Amazon ECS e AWS Lambda, podem acessar um sistema de arquivos Amazon EFS ao mesmo tempo. Portanto, um sistema de arquivos EFS pode fornecer uma fonte de dados comum para workloads e aplicativos em execução em mais de uma instância de computação ou servidor.

Instalando **`amazon-efs-utils`** em AL2 023

O `amazon-efs-utils` pacote está disponível nos repositórios AL2 023 para serem instalados e usados para acessar os sistemas de arquivos do Amazon EFS.

Instale o **`amazon-efs-utils`** pacote em AL2 023

- Instale `amazon-efs-utils` usando o comando a seguir.

```
$ dnf -y install amazon-efs-utils
```

Montando um sistema de arquivos Amazon EFS em AL2 023

Depois `amazon-efs-utils` de instalado, você pode montar um sistema de arquivos Amazon EFS na sua instância AL2 023.

Monte um sistema de arquivos Amazon EFS em AL2 023

- Para montar usando o ID do sistema de arquivos, use o comando a seguir.

```
sudo mount -t efs file-system-id efs-mount-point/
```

Você também pode montar o sistema de arquivos para que os dados em trânsito sejam criptografados usando TLS ou usando o nome DNS ou o IP de destino de montagem em vez do ID do sistema de arquivos. Para obter mais informações, consulte Montagem em instâncias Amazon Linux usando o auxiliar de montagem EFS.

Usando o Amazon EMR criado em 023 AL2

O Amazon EMR é um serviço da web que facilita o processamento de grandes quantidades de dados de maneira eficiente usando o Apache Hadoop e os serviços oferecidos pela AWS.

AL2Lançamentos do Amazon EMR baseados em 023

A versão 7.0.0 do Amazon EMR foi a primeira versão criada em 023. AL2 Com esta versão, o AL2 023 é o sistema operacional básico do Amazon EMR, trazendo todas as vantagens AL2 do 023 para o Amazon EMR. Para obter mais informações, consulte as notas de versão do Amazon EMR 7.0.0.

AL2Amazon EMR baseado em 023 no EKS

O Amazon EMR no EKS 6.13 foi a primeira versão introduzindo a versão AL2 0.23 como opção. Com essa versão, você pode iniciar o Spark com AL2 023 como sistema operacional, junto com o tempo de execução do Java 17. Para obter mais informações, consulte as notas de lançamento do Amazon EMR no EKS 6.13 e todas as notas de lançamento do Amazon EMR no EKS.

Usando AL2 0,23 em AWS Lambda

Com AWS Lambda, você pode executar código sem provisionar ou gerenciar servidores. Você paga somente pelo tempo de computação que consome. Não há cobrança quando seu código não está em execução. Você pode executar código para praticamente qualquer tipo de aplicativo ou serviço de back-end, tudo sem nenhuma administração. Carregar seu código e o Lambda cuidará de tudo que for necessário para executar e escalar seu código com alta disponibilidade.

AL2023 tempo de execução **`provided.al2023`** gerenciado e imagem de contêiner

O tempo de execução `provided.al2023` básico é baseado na imagem mínima do contêiner AL2 023 e fornece um tempo de execução gerenciado do Lambda baseado em AL2 023 e uma imagem base do contêiner. Como o `provided.al2023` tempo de execução é baseado na imagem mínima do contêiner AL2 023, ele é substancialmente menor, com menos de 40 MB, do que o `provided.al2` tempo de execução, com cerca de 109 MB.

Para obter mais informações, consulte Tempos de execução do Lambda e Como trabalhar com imagens de contêiner do Lambda.

AL2Tempos de execução Lambda baseados em 023

Versões futuras de tempos de execução de linguagem gerenciada, como Node.js 20, Python 3.12, Java 21, e .NET 8, são baseados em AL2 023 e serão usados `provided.al2023` como imagem base conforme descrito no anúncio de tempos de execução baseados em AL2 023.

AL2Funções Lambda baseadas em 023

- AL2023 Funções Lambda escritas em Go
- AL2023 Funções Lambda escritas em Rust

Para obter mais informações, consulte Cotas do Lambda no AWS Lambda Guia do desenvolvedor do e.

Tutoriais

Os tutoriais a seguir mostram como realizar tarefas comuns usando EC2 instâncias da Amazon executando o Amazon Linux 2023 (AL2023). Para tutoriais em vídeo, consulte Vídeos AWS instrucionais e laboratórios.

Para AL2 obter instruções, consulte Tutoriais para EC2 instâncias da Amazon executando Linux no Guia EC2 do usuário da Amazon.

Tutoriais

- Tutorial: instalar um servidor LAMP em AL2 023
- Tutorial: Configurar SSL/TLS em 023 AL2
- Tutorial: Hospede um WordPress blog em AL2 023
- Tutorial: Transição do Redis 6 para o Valkey em 023 AL2
- Tutorial: Instale o ambiente de trabalho GNOME em 023 AL2
- Tutorial: Configurar o servidor TigerVNC em 023 AL2

Tutorial: instalar um servidor LAMP em AL2 023

Os procedimentos a seguir ajudam você a instalar um servidor web Apache com suporte a PHP e MariaDB (um fork do MySQL desenvolvido pela comunidade) em AL2 sua instância 023 (às vezes chamada de servidor web LAMP ou pilha LAMP). Você pode usar esse servidor para hospedar um site estático ou para implantar um aplicativo PHP dinâmico que lê e grava informações em um banco de dados.

> **Important**
>
> Esses procedimentos são destinados ao uso com AL2 023. Se você estiver tentando configurar um servidor web LAMP em uma distribuição diferente, como Ubuntu ou Red Hat Enterprise Linux, este tutorial não funcionará. Para o Ubuntu, consulte a seguinte documentação da comunidade Ubuntu: ApacheMySQLPHP. Para outras distribuições, consulte a documentação específica.

Tarefas

Etapa 1: Preparar o servidor LAMP

Pré-requisitos

- Este tutorial pressupõe que você já tenha iniciado uma nova instância usando AL2 023, com um nome DNS público que pode ser acessado pela Internet. Para obter mais informações, consulte AL2023 na Amazon EC2. Você também precisa ter configurado o security group para permitir conexões SSH (porta 22), HTTP (porta 80) e HTTPS (porta 443). Para obter mais informações sobre esses pré-requisitos, consulte Autorizar tráfego de entrada para suas instâncias Linux no Guia do usuário da Amazon. EC2
- O procedimento a seguir instala a versão mais recente do PHP disponível em AL2 023, atualmente 8.1. Se você planeja usar aplicações PHP diferentes daquelas descritas neste tutorial, você deve verificar a compatibilidade com a versão 8.1.

Para preparar o servidor LAMP

1. Conecte-se à sua instância. Para obter mais informações, consulte Conexão com AL2 203 instâncias.
2. Para garantir que todos os pacotes de software estejam atualizados, execute uma atualização rápida de software em sua instância. Esse processo poderá levar alguns minutos, mas é importante ter certeza de que você tem as atualizações de segurança e correções de bug mais recentes.

   A opção `-y` instala as atualizações sem solicitar confirmação. Para examinar as atualizações antes da instalação, você pode omitir essa opção.

   ```
   [ec2-user ~]$ sudo dnf upgrade -y
   ```
3. Instale as versões mais recentes do servidor web Apache e dos pacotes PHP para AL2 0.23.

```
[ec2-user ~]$ sudo dnf install -y httpd wget php-fpm php-mysqli php-json php php-
devel
```

4. Instale os pacotes de software do MariaDB. Use o comando dnf install para instalar os vários pacotes de software e todas as dependências relacionadas ao mesmo tempo.

```
[ec2-user ~]$ sudo dnf install mariadb105-server
```

Você pode visualizar as versões atuais desses pacotes usando o comando a seguir:

```
[ec2-user ~]$ sudo dnf info package_name
```

Exemplo: .

```
[root@ip-172-31-25-170 ec2-user]# dnf info mariadb105
Last metadata expiration check: 0:00:16 ago on Tue Feb 14 21:35:13 2023.
Installed Packages
Name         : mariadb105
Epoch        : 3
Version      : 10.5.16
Release      : 1.amzn2023.0.6
Architecture : x86_64
Size         : 18 M
Source       : mariadb105-10.5.16-1.amzn2023.0.6.src.rpm
Repository   : @System
From repo    : amazonlinux
Summary      : A very fast and robust SQL database server
URL          : http://mariadb.org
License      : GPLv2 and LGPLv2
Description  : MariaDB is a community developed fork from MySQL - a multi-user,
 multi-threaded
             : SQL database server. It is a client/server implementation consisting
 of
             : a server daemon (mariadbd) and many different client programs and
 libraries.
             : The base package contains the standard MariaDB/MySQL client programs
 and
             : utilities.
```

5. Inicie o servidor web Apache.

```
[ec2-user ~]$ sudo systemctl start httpd
```

6. Use o comando systemctl para configurar o servidor web Apache para iniciar em cada inicialização do sistema.

   ```
   [ec2-user ~]$ sudo systemctl enable httpd
   ```

   Você pode verificar se httpd está ativo executando o seguinte comando:

   ```
   [ec2-user ~]$ sudo systemctl is-enabled httpd
   ```

7. Adicione uma regra de segurança para permitir conexões HTTP de entrada (porta 80) na instância caso você ainda não tenha feito isso. Por padrão, um grupo de *N* segurança do assistente de inicialização foi criado para sua instância durante a execução. Se você não acrescentou regras adicionais de grupo de segurança, esse grupo contém apenas uma única regra para permitir conexões SSH.

   a. Abra o EC2 console da Amazon em https://console.aws.amazon.com/ec2/.

   b. No navegador à esquerda, selecione Instances (Instâncias) e selecione sua instância.

   c. Na guia Security (Segurança), exiba as regras de entrada. Você deve ver a seguinte regra:

      ```
      Port range   Protocol     Source
      22           tcp          0.0.0.0/0
      ```

      > **Warning**
      >
      > Usar `0.0.0.0/0` permite que todos os IPv4 endereços acessem sua instância usando SSH. Isso é aceitável para um período curto em um ambiente de teste, mas não é seguro em ambientes de produção. Na produção, você autorizará somente um endereço IP específico ou intervalo de endereços para acessar a instância.

   d. Se não houver uma regra de entrada para permitir conexões HTTP (porta 80), será necessário adicionar a regra agora. Escolha o link do grupo de segurança. Usando os procedimentos em Autorizar tráfego de entrada para suas instâncias Linux, adicione uma nova regra de segurança de entrada com os seguintes valores:

      - Tipo: HTTP

- Protocolo: TCP
- Port Range: 80
- Source (Origem): personalizado

8. Teste o servidor web. Em um navegador, digite o endereço DNS público (ou o endereço IP público) de sua instância. Se não houver conteúdo em `/var/www/html`, você deverá verificar a página de teste do Apache, que exibirá a mensagem “It works!” (Funciona!).

   Você pode obter o DNS público da sua instância usando o EC2 console da Amazon (verifique a coluna IPv4 DNS público; se essa coluna estiver oculta, escolha Preferências (o ícone em forma de engrenagem) e ative DNS público). IPv4

   Verifique se o grupo de segurança da instância contém uma regra para permitir o tráfego HTTP na porta 80. Para obter mais informações, consulte Adicionar regras ao grupo de segurança.

   > **Important**
   >
   > Se você não estiver usando o Amazon Linux, talvez seja necessário configurar o firewall na instância para permitir essas conexões. Para obter mais informações sobre como configurar o firewall, consulte a documentação de sua distribuição específica.

O httpd do Apache é usado para os arquivos que são mantidos em um diretório chamado raiz de documentos do Apache. O diretório raiz de documentos Apache do Amazon Linux é `/var/www/html`, que, por padrão, é de propriedade da raiz.

Para permitir que a conta do `ec2-user` manipule arquivos nesse diretório, você deve modificar a propriedade e as permissões do diretório. Existem diversas maneiras de realizar essa tarefa. Neste tutorial, você adiciona `ec2-user` ao grupo `apache` para dar ao grupo `apache` a propriedade do diretório `/var/www` e atribuir permissões de gravação ao grupo.

Para definir permissões de arquivo

1. Adicione o usuário (neste caso, o `ec2-user`) ao grupo do `apache`.

   ```
   [ec2-user ~]$ sudo usermod -a -G apache ec2-user
   ```

2. Faça logout e login novamente para selecionar o novo grupo verifique sua associação.

   a. Faça logout (use o comando exit ou feche a janela do terminal):

```
[ec2-user ~]$ exit
```

b. Para verificar sua associação no grupo `apache`, reconecte-se à instância e execute o comando a seguir:

```
[ec2-user ~]$ groups
ec2-user adm wheel apache systemd-journal
```

3. Altere a propriedade do grupo do `/var/www` e seu conteúdo para o grupo do `apache`.

```
[ec2-user ~]$ sudo chown -R ec2-user:apache /var/www
```

4. Para adicionar as permissões de gravação do grupo e definir o ID do grupo nos subdiretórios futuros, altere as permissões de diretório de `/var/www` e de seus subdiretórios.

```
[ec2-user ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod
 2775 {} \;
```

5. Para adicionar permissões de gravação do grupo, altere recursivamente as permissões de arquivo de `/var/www` e de seus subdiretórios:

```
[ec2-user ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
```

Agora, `ec2-user` (e outros todos os futuros do grupo `apache`) poderão adicionar, excluir e editar arquivos na raiz do documento Apache, permitindo que você adicione conteúdo, como um site estático ou uma aplicação PHP.

Para proteger o servidor web (opcional)

Um servidor web que executa o protocolo HTTP não fornece nenhuma segurança de transporte para os dados que envia ou recebe. Quando você se conecta a um servidor HTTP usando um navegador da Web, o conteúdo URLs que você visita, o conteúdo das páginas da Web que você recebe e o conteúdo (incluindo senhas) de qualquer formulário HTML que você envia são todos visíveis para espiões em qualquer lugar ao longo do caminho da rede. A melhor prática para proteger o servidor web é instalar suporte para HTTPS (HTTP seguro), que protege os dados por meio de criptografia SSL/TLS.

Para obter informações sobre como habilitar o HTTPS no servidor, consulte Tutorial: Configurar SSL/TLS em 023 AL2.

Etapa 2: Testar o servidor LAMP

Se o servidor estiver instalado e em execução, e suas permissões de arquivo estiverem definidas corretamente, a conta do `ec2-user` poderá criar um arquivo PHP no diretório `/var/www/html` disponível na Internet.

Para testar o servidor do LAMP

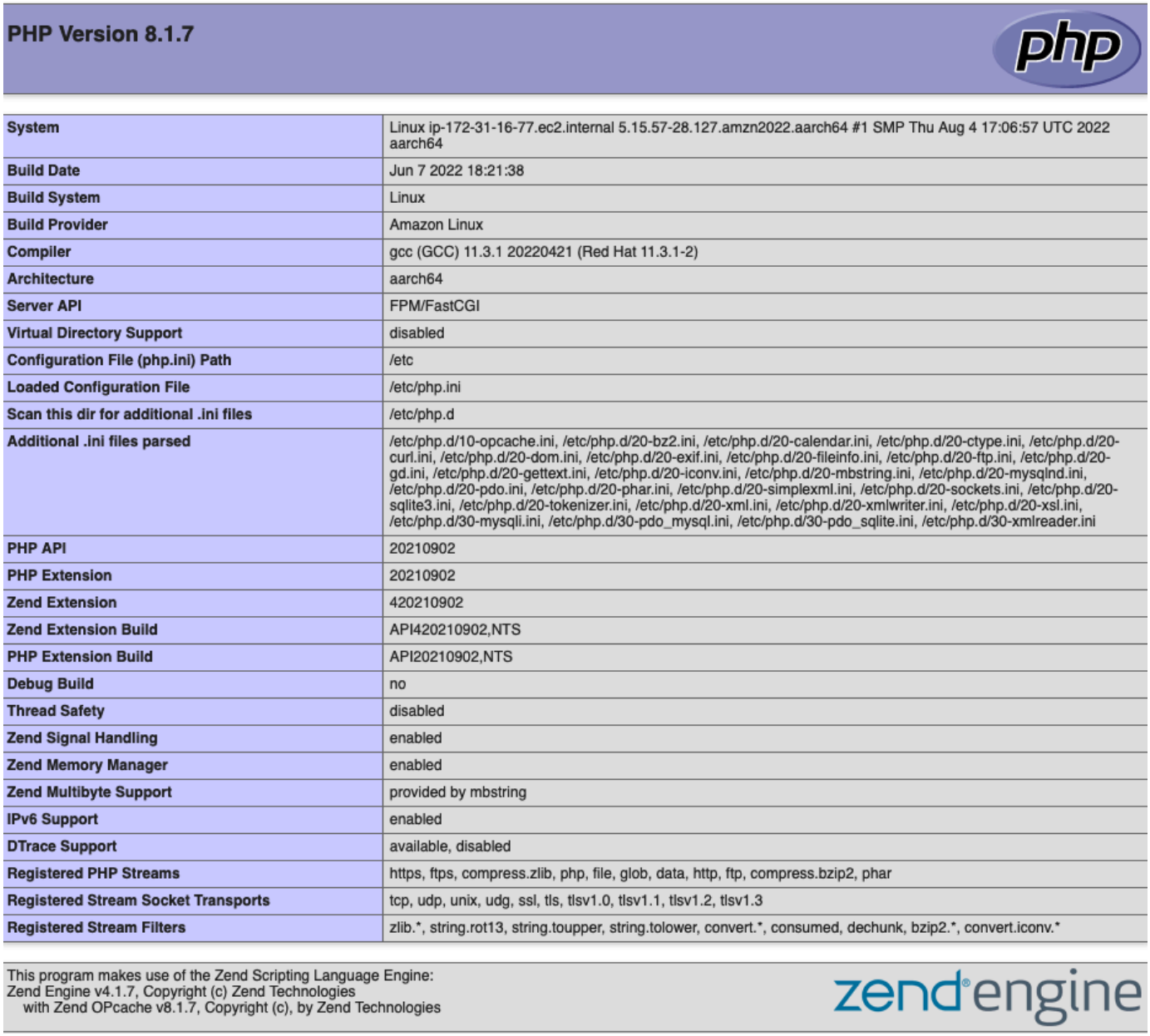
1. Crie um arquivo PHP no diretório base do Apache.

   ```
   [ec2-user ~]$ echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
   ```

   Se você receber o erro ”Permissão negada" ao tentar executar esse comando, tente fazer logout e login novamente para obter as permissões corretas do grupo que você configurou em Para definir permissões de arquivo.
2. Em um navegador da web, digite a URL do arquivo que você acabou de criar. Essa URL é o endereço DNS público da instância seguido por uma barra e o nome do arquivo. Por exemplo:

   ```
   http://my.public.dns.amazonaws.com/phpinfo.php
   ```

   Você deve ver a página de informações do PHP:



PHP Version 8.1.7

System	Linux ip-172-31-16-77.ec2.internal 5.15.57-28.127.amzn2022.aarch64 #1 SMP Thu Aug 4 17:06:57 UTC 2022 aarch64
Build Date	Jun 7 2022 18:21:38
Build System	Linux
Build Provider	Amazon Linux
Compiler	gcc (GCC) 11.3.1 20220421 (Red Hat 11.3.1-2)
Architecture	aarch64
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/10-opcache.ini, /etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-dom.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gd.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-mbstring.ini, /etc/php.d/20-mysqlnd.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-simplexml.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, /etc/php.d/20-xmlwriter.ini, /etc/php.d/20-xsl.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini, /etc/php.d/30-pdo_sqlite.ini, /etc/php.d/30-xmlreader.ini
PHP API	20210902
PHP Extension	20210902
Zend Extension	420210902
Zend Extension Build	API420210902,NTS
PHP Extension Build	API20210902,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, compress.bzip2, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, bzip2.*, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:
Zend Engine v4.1.7, Copyright (c) Zend Technologies
with Zend OPcache v8.1.7, Copyright (c), by Zend Technologies

Se você não vir essa página, verifique se o arquivo `/var/www/html/phpinfo.php` foi criado corretamente na etapa anterior. Você também pode verificar se todos os pacotes necessários foram instalados com o comando a seguir.

```
[ec2-user ~]$ sudo dnf list installed httpd mariadb-server php-mysqlnd
```

Se alguns dos pacotes necessários não estiverem listados na saída, instale-os com o comando sudo yum install ***package***.

3. Exclua o arquivo `phpinfo.php`. Embora essas informações possam ser úteis, elas não devem ser transmitidas pela Internet por motivos de segurança.

```
[ec2-user ~]$ rm /var/www/html/phpinfo.php
```

Agora você deve ter um servidor web do LAMP totalmente funcional. Se adicionar conteúdo ao diretório base do Apache em `/var/www/html`, você deverá poder visualizar esse conteúdo no endereço DNS público de sua instância.

Etapa 3: Proteger o servidor do banco de dados

A instalação padrão do servidor MariaDB tem vários recursos que são bons para teste e desenvolvimento, mas devem ser desabilitados ou removidos em servidores de produção. O comando mysql_secure_installation orienta você durante o processo de configuração de uma senha raiz e da remoção de recursos não seguros da instalação. Mesmo que você não esteja planejando usar o servidor MariaDB é recomendável executar este procedimento.

Para proteger o servidor MariaDB

1. Inicie o servidor MariaDB.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

2. Executar mysql_secure_installation.

```
[ec2-user ~]$ sudo mysql_secure_installation
```

   a. Quando solicitado, digite uma senha para a conta raiz.

      i. Digite a senha raiz atual. Por padrão, a conta raiz não tem uma senha definida. Pressione Enter.

      ii. Digite **Y** para definir uma senha e digite uma senha segura duas vezes. Para obter mais informações sobre como criar uma senha segura, consulte https://identitysafe.norton.com/password-generator/. Armazene essa senha em um lugar seguro.

      A configuração de uma senha raiz para o MariaDB é somente a medida mais básica para proteger seu banco de dados. Ao criar ou instalar um aplicativo controlado por

banco de dados, geralmente, você cria um usuário de serviço de banco para esse aplicativo e evita usar a conta raiz para qualquer coisa que não seja a administração do banco de dados.

b. Digite **Y** para remover as contas de usuários anônimos.

c. Digite **Y** para desabilitar o recurso de login remoto da raiz.

d. Digite **Y** para remover o banco de dados de teste.

e. Digite **Y** para recarregar as tabelas de privilégios e salvar suas alterações.

3. (Opcional) Se você não pretende usar o servidor MariaDB imediatamente, interrompa-o. Você poderá reiniciá-lo quando precisar dele novamente.

```
[ec2-user ~]$ sudo systemctl stop mariadb
```

4. (Opcional) Se você quiser que o servidor MariaDB seja iniciado a cada inicialização, digite o comando a seguir.

```
[ec2-user ~]$ sudo systemctl enable mariadb
```

Etapa 4: Instalação (opcional) phpMyAdmin

phpMyAdminé uma ferramenta de gerenciamento de banco de dados baseada na web que você pode usar para visualizar e editar os bancos de dados MySQL na EC2 sua instância. Siga as etapas a seguir para instalar e configurar o `phpMyAdmin` em sua instância do Amazon Linux.

> **Important**
>
> Não recomendamos usar o `phpMyAdmin` para acessar um servidor LAMP, a menos que você tenha habilitado o SSL/TLS no Apache. Caso contrário, sua senha de administrador de banco de dados e outros dados serão transmitidos de forma desprotegida pela Internet. Para obter recomendações de segurança dos desenvolvedores, consulte Protegendo sua phpMyAdmin instalação. Para obter informações gerais sobre como proteger um servidor web em uma EC2 instância, consulteTutorial: Configurar SSL/TLS em 023 AL2.

Para instalar phpMyAdmin

1. Instale as dependências necessárias.

```
[ec2-user ~]$ sudo dnf install php-mbstring php-xml -y
```

2. Reinicie o Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

3. Reinicie o `php-fpm`.

```
[ec2-user ~]$ sudo systemctl restart php-fpm
```

4. Navegue até o diretório base do Apache em `/var/www/html`.

```
[ec2-user ~]$ cd /var/www/html
```

5. Selecione um pacote de origem para a phpMyAdmin versão mais recente em https://www.phpmyadmin.net/downloads. Para fazer download do arquivo diretamente para a instância, copie o link e cole-o em um comando wget, como neste exemplo:

```
[ec2-user html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
```

6. Crie uma pasta `phpMyAdmin` e extraia o pacote dela com o comando a seguir.

```
[ec2-user html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1
```

7. Exclua o *`phpMyAdmin-latest-all-languages.tar.gz`* tarball.

```
[ec2-user html]$ rm phpMyAdmin-latest-all-languages.tar.gz
```

8. (Opcional) Se o servidor MySQL não estiver em execução, inicie-o agora.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

9. Em um navegador da Web, digite a URL da sua phpMyAdmin instalação. Essa URL é o endereço DNS público (ou o endereço IP público) da instância seguido por uma barra e o nome do diretório de instalação. Por exemplo:

```
http://my.public.dns.amazonaws.com/phpMyAdmin
```

Você deve ver a página de phpMyAdmin login:

10. Faça login na sua phpMyAdmin instalação com o nome de `root` usuário e a senha raiz do MySQL que você criou anteriormente.

    A instalação ainda deve ser configurada antes que você a coloque em serviço. Sugerimos que você comece criando manualmente o arquivo de configuração, da seguinte maneira:

    a. Para começar com um arquivo de configuração mínima, use seu editor de texto favorito para criar um novo arquivo e, em seguida, copie o conteúdo de `config.sample.inc.php` para ele.

    b. Salve o arquivo como `config.inc.php` no phpMyAdmin diretório que contém`index.php`.

    c. Consulte as instruções de criação pós-arquivo na seção Usando o script de phpMyAdmin instalação das instruções de instalação para qualquer configuração adicional.

Para obter informações sobre o uso phpMyAdmin, consulte o Guia phpMyAdmin do usuário.

Solução de problemas

Esta seção oferece sugestões para resolver problemas comuns que podem surgir durante a configuração de um novo servidor do LAMP.

Não consigo me conectar ao servidor usando um navegador da web

Execute as seguintes verificações para ver se o servidor da web do Apache está em execução e acessível.

- O servidor web está em execução?

  Você pode verificar se httpd está ativo executando o seguinte comando:

  ```
  [ec2-user ~]$ sudo systemctl is-enabled httpd
  ```

  Se o processo httpd não estiver em execução, repita as etapas descritas em Para preparar o servidor LAMP.
- O firewall está configurado corretamente?

  Verifique se o grupo de segurança da instância contém uma regra para permitir o tráfego HTTP na porta 80. Para obter mais informações, consulte Adicionar regras ao grupo de segurança.

Não consigo me conectar ao meu servidor usando HTTPS

Execute as seguintes verificações para ver se o servidor da web do Apache está configurado para dar suporte a HTTPS.

- O servidor Web está configurado corretamente?

  Depois de instalar o Apache, o servidor é configurado para tráfego HTTP. Para suportar HTTPS, ative o TLS no servidor e instale um certificado SSL. Para ter mais informações, consulte Tutorial: Configurar SSL/TLS em 023 AL2.
- O firewall está configurado corretamente?

Verifique se o grupo de segurança da instância contém uma regra para permitir o tráfego HTTPS na porta 443. Para obter mais informações, consulte Autorizar tráfego de entrada para suas instâncias Linux.

Tópicos relacionados

Para obter mais informações sobre como transferir arquivos para sua instância ou instalar um WordPress blog em seu servidor web, consulte a documentação a seguir:

- Transfira arquivos para sua instância Linux usando o WinSCP no Amazon EC2 User Guide.
- Transfira arquivos para instâncias Linux usando um cliente SCP no Amazon EC2 User Guide.
- Tutorial: Hospede um WordPress blog em AL2 023

Para obter mais informações sobre os comandos e o software usados neste tutorial, consulte as seguintes páginas da web:

- Servidor web Apache: http://httpd.apache.org/
- Servidor de banco de dados MariaDB: https://mariadb.org/
- Linguagem de programação PHP: http://php.net/

Para obter mais informações sobre como registrar um nome de domínio para o servidor web ou transferir um nome de domínio existente para este host, consulte Como criar e migrar domínios e subdomínios para o Amazon Route 53 no Guia do desenvolvedor do Amazon Route 53.

Tutorial: Configurar SSL/TLS em 023 AL2

Soquetes seguros Layer/Transport Layer Security (SSL/TLS) creates an encrypted channel between a web server and web client that protects data in transit from being eavesdropped on. This tutorial explains how to add support manually for SSL/TLS em uma EC2 instância com AL2 023 e servidor web Apache. Este tutorial pressupõe que você não esteja usando um balanceador de carga. Se você estiver usando Elastic Load Balancing, poderá optar por configurar o descarregamento do SSL no balanceador de carga, usando, em vez disso, um certificado do AWS Certificate Manager.

Por motivos históricos, a criptografia na Web é conhecida simplesmente como SSL. Embora os navegadores da Web ainda suportem SSL, seu protocolo sucessor, o TLS, é menos vulnerável

a ataques. AL2023 desativa o suporte do lado do servidor para todas as versões do SSL por padrão. Órgãos de normas de segurança consideram o TLS 1.0 não seguro. O TLS 1.0 e TLS 1.1 foram formalmente preteridos em março de 2021. Este tutorial contém orientações baseadas exclusivamente na ativação do TLS 1.2. O TLS 1.3 foi finalizado em 2018 e está disponível AL2 desde que a biblioteca TLS subjacente (OpenSSL neste tutorial) seja suportada e habilitada. Os clientes devem ser compatíveis com o TLS 1.2 ou posterior até 28 de junho de 2023. Para obter mais informações sobre os padrões de criptografia atualizados, consulte RFC 7568 e RFC 8446.

Este tutorial refere-se à criptografia da Web moderna simplesmente como TLS.

Important

Esses procedimentos são destinados ao uso com AL2 023. Se você estiver tentando configurar uma EC2 instância executando uma distribuição diferente ou uma instância executando uma versão antiga do Amazon Linux, alguns procedimentos deste tutorial podem não funcionar. Para o Ubuntu, consulte a seguinte documentação da comunidade do Ubuntu: Open SSL on Ubuntu (Open SSL no Ubuntu). Para o Red Hat Enterprise Linux, consulte: Como configurar o Servidor Web Apache HTTP. Para outras distribuições, consulte a documentação específica.

Note

Como alternativa, você pode usar o AWS Certificate Manager (ACM) para enclaves AWS Nitro, que é um aplicativo de enclave que permite usar certificados SSL/TLS públicos e privados com seus aplicativos e servidores web em execução em instâncias da Amazon com o Nitro Enclaves. EC2 AWS O Nitro Enclaves é um EC2 recurso da Amazon que permite a criação de ambientes computacionais isolados para proteger e processar com segurança dados altamente confidenciais, como certificados SSL/TLS e chaves privadas.
O ACM for Nitro Enclaves funciona com o nginx em execução na sua instância Amazon EC2 Linux para criar chaves privadas, distribuir certificados e chaves privadas e gerenciar renovações de certificados.
Para usar o ACM for Nitro Enclaves, é necessário usar uma instância do Linux habilitada para enclave.
Para obter mais informações, consulte O que são AWS Nitro Enclaves? e AWS Certificate Manager para Nitro Enclaves no Guia do usuário do AWS Nitro Enclaves.

Conteúdo

Pré-requisitos

Antes de começar este tutorial, conclua as seguintes etapas:

- Execute uma instância AL2 023 com suporte do EBS. Para obter mais informações, consulte AL2023 na Amazon EC2.
- Configure seus grupos de segurança para permitir que sua instância aceite conexões nas seguintes portas TCP:
  - SSH (porta 22)
  - HTTP (porta 80)
  - HTTPS (porta 443)

  Para obter mais informações, consulte Autorizar tráfego de entrada para suas instâncias Linux no Guia EC2 do usuário da Amazon.
- Instale o servidor Web Apache. Para step-by-step obter instruções, consulteTutorial: instalar um servidor LAMP em AL2 023. Somente o pacote httpd e suas dependências são necessários e, portanto, você pode ignorar as instruções que envolvem PHP e MariaDB.
- Para identificar e autenticar sites, a infraestrutura de chave pública (PKI) do TLS depende do Sistema de Nomes de Domínio (DNS). Para usar sua EC2 instância para hospedar um site público, você precisa registrar um nome de domínio para seu servidor web ou transferir um nome de domínio existente para seu EC2 host da Amazon. Há vários serviços de registro de domínio e de hospedagem DNS de terceiros disponíveis para isso, ou você pode usar o Amazon Route 53.

Etapa 1: habilitar o TLS no servidor

Esse procedimento conduz você pelo processo de configuração do TLS no AL2 023 com um certificado digital autoassinado.

> **Note**
>
> Um certificado autoassinado é aceitável para testes, mas não para produção. Quando você expõe seu certificado autoassinado na Internet, os visitantes de seu site recebem avisos de segurança.

Para habilitar o TLS em um servidor

1. Conecte-se à sua instância e confirme se o Apache está em execução. Para obter mais informações, consulte Conexão com AL2 203 instâncias.

   ```
   [ec2-user ~]$ sudo systemctl is-enabled httpd
   ```

   Se o valor retornado não for "habilitado", inicie o Apache e configure-o para iniciar sempre que o sistema for inicializado.

   ```
   [ec2-user ~]$ sudo systemctl start httpd && sudo systemctl enable httpd
   ```

2. Para garantir que todos os pacotes de software estejam atualizados, execute uma atualização rápida de software em sua instância. Esse processo pode levar alguns minutos, mas é importante ter certeza de que você tem as atualizações de segurança e correções de bug mais recentes.

   > **Note**
   >
   > A opção `-y` instala as atualizações sem solicitar confirmação. Para examinar as atualizações antes da instalação, você pode omitir essa opção.

   ```
   [ec2-user ~]$ sudo dnf install openssl mod_ssl
   ```

3. Depois de inserir o seguinte comando, você será levado a um prompt onde poderá inserir informações sobre seu site.

   ```
   [ec2-user ~]$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /
   etc/pki/tls/private/apache-selfsigned.key -out /etc/pki/tls/certs/apache-
   selfsigned.crt
   ```

Isso gera um novo arquivo `apache-selfsigned.crt` no diretório `/etc/pki/tls/certs/`. O nome do arquivo especificado corresponde ao padrão atribuído na diretiva SSLCertificateFile em `/etc/httpd/conf.d/ssl.conf`.

Sua instância agora possui os seguintes arquivos que você usará para configurar seu servidor seguro e criar um certificado para teste:

- `/etc/httpd/conf.d/ssl.conf`

  O arquivo de configuração para mod_ssl. Contém as diretrizes que informam ao Apache onde encontrar chaves de criptografia e certificados, as versões do protocolo TLS a serem permitidas e as cifras de criptografia a serem aceitas. Este será o seu arquivo de certificado local:

- `/etc/pki/tls/certs/apache-selfsigned.crt`

Esse arquivo contém um certificado autoassinado e a chave privada do certificado. O Apache requer que o certificado e a chave estejam no formato PEM, que consiste em caracteres ASCII codificados em Base64 enquadrados pelas linhas "BEGIN" e "END", como neste exemplo.

```
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQD2KKx/8Zk94m1q
3gQMZF9ZN66Ls19+3tHAgQ5Fpo9KJDhzLjOOCI8u1PTcGmAah5kEitCEc0wzmNeo
BCl0wYR6G0rGaKtK9Dn7CuIjvubtUysVyQoMVPQ97ldeakHWeRMiEJFXg6kZZ0vr
GvwnKoMh3DlK44D9dX7IDua2PlYx5+eroA+1Lqf32ZSaAO0bBIMIYTHigwbHMZoT
...
56tE7THvH7vOEf4/iUOsIrEzaMaJ0mqkmY1A70qQGQKBgBF3H1qNRNHuyMcPODFs
27hDzPDinrquSEvoZIggkDMlh2irTiipJ/GhkvTpoQlv0fK/VXw8vSgeaBuhwJvS
LXU9HvYq0U6O4FgD3nAyB9hI0BE13r1HjUvbjT7moH+RhnNz6eqqdscCS09VtRAo
4QQvAqOa8UheYeoXLdWcHaLP
-----END PRIVATE KEY-----

-----BEGIN CERTIFICATE-----
MIIEazCCA1OgAwIBAgICWxQwDQYJKoZIhvcNAQELBQAwgbExCzAJBgNVBAYTAi0t
MRIwEAYDVQQIDAlTb21lU3RhdGUxETAPBgNVBAcMCFNvbWVDaXR5MRkwFwYDVQQK
DBBTb21lT3JnYW5pemF0aW9uMR8wHQYDVQQLDBZTb21lT3JnYW5pemF0aW9uYWxV
bml0MRkwFwYDVQQDDBBpcC0xNzItMzEtMjAtMjM2MSQwIgYJKoZIhvcNAQkBFhVy
...
z5rRUE/XzxRLBZOoWZpNWTXJkQ3uFYH6s/sBwtHpKKZMzOvDedREjNKAvk4ws6F0
CuIjvubtUysVyQoMVPQ97ldeakHWeRMiEJFXg6kZZ0vrGvwnKoMh3DlK44D9dlU3
WanXWehT6FiSZvB4sTEXXJN2jdw8g+sHGnZ8zCOsclknYhHrCVD2vnBlZJKSZvak
```

```
3ZazhBxtQSukFMOnWPP2a0DMMFGYUHOd0BQE8sBJxg==
-----END CERTIFICATE-----
```

Os nomes de arquivos e as extensões são uma conveniência e não têm efeito na função. Por exemplo, você pode chamar um certificado de `cert.crt`, `cert.pem` ou de um outro nome de arquivo qualquer, desde que a diretiva relacionada no arquivo `ssl.conf` use o mesmo nome.

> **Note**
>
> Ao substituir os arquivos TLS padrão por seus próprios arquivos personalizados, verifique se eles estão no formato PEM.

4. Reinicie o Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

> **Note**
>
> Certifique-se de que a porta TCP 443 esteja acessível na sua EC2 instância, conforme descrito anteriormente.

5. Seu servidor da Web do Apache agora deve oferecer suporte a HTTPS (HTTP seguro) por meio da porta 443. Teste inserindo o endereço IP ou o nome de domínio totalmente qualificado da sua EC2 instância em uma barra de URL do navegador com o prefixo**`https://`**.

   Como você está se conectando a um site com um certificado de host autoassinado não confiável, o navegador poderá exibir uma série de avisos de segurança. Ignore os avisos e continue para o site.

   Se a página de teste padrão do Apache for aberta, a configuração do TLS no servidor estará correta. Todos os dados que passam entre o navegador e o servidor agora estão criptografados.

> **Note**
>
> Para impedir que os visitantes do site encontrem telas de avisos, você precisa obter um certificado assinado por uma CA confiável que, além de criptografar, também autentique você publicamente como o proprietário do site.

Etapa 2: obter um certificado assinado por uma CA

Você pode seguir este processo para obter um certificado assinado por uma CA:

- Gere uma solicitação de assinatura de certificado (CSR) a partir de uma chave privada
- Enviar a CSR para uma autoridade de certificação (CA)
- Obtenha um certificado de host assinado
- Configure o Apache para usá-lo

Um certificado de host TLS X.509 autoassinado é idêntico em termos criptológicos a um certificado assinado por uma CA. A diferença é social, não matemática. Uma CA promete validar, no mínimo, a propriedade de um domínio antes de emitir um certificado para um candidato. Cada navegador da Web contém uma lista de CAs informações confiáveis do fornecedor do navegador para fazer isso. Primariamente, um certificado X.509 consiste em uma chave pública, que corresponde à chave privada do servidor, e uma assinatura pela CA que é vinculada criptograficamente à chave pública. Quando um navegador se conecta a um servidor da Web por HTTPS, o servidor apresenta um certificado para o navegador verificar em relação à sua lista de confiáveis CAs. Se o assinante estiver na lista ou for acessível por meio de uma cadeia de confiança que consiste em outros assinantes confiáveis, o navegador negociará um canal rápido de dados criptografados com o servidor e carregará a página.

Geralmente, os certificados são caros devido ao trabalho envolvido na validação das solicitações, portanto, vale a pena comparar os preços. Alguns CAs oferecem certificados de nível básico gratuitamente. O mais notável deles CAs é o projeto Let's Encrypt, que também suporta a automação do processo de criação e renovação de certificados. Para obter mais informações sobre como usar um certificado Let’s Encrypt, consulte Obtenção do Certbot.

Se você planeja oferecer serviços de nível comercial, o AWS Certificate Manager é uma boa opção.

É importante ter um certificado de host subjacente. Desde 2019, grupos governamentais e do setor recomendam usar um tamanho de chave (módulo) mínimo de 2.048 bits para chaves de RSA para a proteção de documentos até 2030. O tamanho do módulo padrão gerado pelo OpenSSL AL2 em 023 é de 2048 bits, o que é adequado para uso em um certificado assinado pela CA. No procedimento a seguir, uma etapa opcional é fornecida para aqueles que desejam uma chave personalizada, por exemplo, uma com módulo maior ou que usa um algoritmo diferente de criptografia.

> **Important**
>
> As instruções para adquirir certificados de host assinados pela CA não funcionarão, a menos que você possua um domínio DNS registrado e hospedado.

Para obter um certificado assinado por uma CA

1. Conecte-se à sua instância e navegue por to /etc/pki/tls/private /. Este é o diretório onde você armazenará a chave privada do servidor para TLS. Se você preferir usar uma chave de host existente para gerar a CSR, vá para a Etapa 3. Para obter mais informações sobre como se conectar à sua instância, consulte Conexão com AL2 203 instâncias
2. (Opcional) Gerar uma nova chave privada. Estes são alguns exemplos de configurações de chave. Qualquer uma das chaves resultantes funciona com seu servidor Web, mas elas variam no grau e no tipo de segurança que elas implementam.

   - Exemplo 1: criar uma chave host de RSA padrão. O arquivo resultante, **`custom.key`**, é uma chave privada de RSA de 2048 bits.

     ```
     [ec2-user ~]$ sudo openssl genrsa -out custom.key
     ```

   - Exemplo 2: criar uma chave de RSA mais forte com um módulo maior. O arquivo resultante, **`custom.key`**, é uma chave privada de RSA de 4096 bits.

     ```
     [ec2-user ~]$ sudo openssl genrsa -out custom.key 4096
     ```

   - Exemplo 3: criar uma chave de RSA de 4096 bits criptografada com proteção por senha. O arquivo resultante, **`custom.key`**, é uma chave privada de RSA de 4096 bits criptografada com a cifra AES-128.

     > **Important**
     >
     > A criptografia da chave fornece maior segurança, mas como uma chave criptografada requer uma senha, os serviços que dependem dela não podem ser iniciados automaticamente. Sempre que usar essa chave, você precisará fornecer a senha (no exemplo anterior, "abcde12345") por meio de uma conexão SSH.

```
[ec2-user ~]$ sudo openssl genrsa -aes128 -passout pass:abcde12345 -out
 custom.key 4096
```

- Exemplo 4: criar uma chave usando uma cifra não RSA. A criptografia RSA pode ser relativamente devagar devido ao tamanho de suas chaves públicas, que são baseadas no produto de dois números primos grandes. No entanto, é possível criar chaves para TLS que usam códigos não RSA. As chaves baseadas em matemática de curvas elípticas são menores e computacionalmente mais rápidas para fornecer um nível de segurança equivalente.

  ```
  [ec2-user ~]$ sudo openssl ecparam -name prime256v1 -out custom.key -genkey
  ```

  O resultado é uma chave privada de curva elíptica de 256 bits que usa prime256v1, uma "curva nomeada" compatível com OpenSSL. A força de criptografia é um pouco maior que uma chave de RSA de 2048 bits, de acordo com o NIST.

  > **Note**
  >
  > Nem todos CAs oferecem o mesmo nível de suporte para elliptic-curve-based chaves que para chaves RSA.

Verifique se a nova chave privada tem a propriedade e permissões altamente restritivas (owner=root, group=root, leitura/gravação para o proprietário somente). O comando será o mostrado no exemplo a seguir.

```
[ec2-user ~]$ sudo chown root:root custom.key
[ec2-user ~]$ sudo chmod 600 custom.key
[ec2-user ~]$ ls -al custom.key
```

Os comandos anteriores produzem o resultado a seguir.

```
-rw------- root root custom.key
```

Depois de criar e configurar uma chave satisfatória, você pode criar uma CSR.

3. Crie uma CSR usando sua chave preferida. O exemplo a seguir usa **`custom.key`**.

```
[ec2-user ~]$ sudo openssl req -new -key custom.key -out csr.pem
```

A OpenSSL abre uma caixa de diálogo e solicita a informação exibida na tabela a seguir. Todos os campos, exceto Common Name (Nome comum), são opcionais para um certificado de host básico validado por domínio.

Nome	Descrição	Exemplo
Nome do país	A abreviação ISO de duas letras para seu país.	US (=Estados Unidos)
Nome do estado ou província	O nome do estado ou província onde sua organização está localizada. Este nome não pode ser abreviado.	Washington
Nome da localidade	A localização de sua organização, como uma cidade.	Seattle
Nome da organização	A razão social completa da sua organização. Não abrevie o nome de sua organização.	Corporação de exemplo
Nome da unidade organizacional	Informações organizacionais adicionais, se houver.	Departamento de exemplo

Nome	Descrição	Exemplo
Nome comum	Esse valor deve corresponder exatamente ao endereço Web que você espera que os usuários digitem em um navegador. Geralmente, isso significa um nome de domínio com um nome de host ou alias prefixados na forma **`www.example.com`** . Em testes com um certificado autoassinado e sem resolução de DNS, o nome comum pode consistir apenas no nome do host. CAs também oferecem certificados mais caros que aceitam nomes curingas, como. **`*.example.com`**	www.exemplo.com
Endereço de e-mail	O endereço de e-mail do administrador do servidor.	someone@example.com

Finalmente, a OpenSSL solicita uma senha de desafio opcional. Essa senha se aplica somente à CSR e às transações entre você e sua CA, portanto, siga as recomendações da CA sobre este e o outro campo opcional, nome da empresa opcional. A senha de desafio da CSR não tem nenhum efeito sobre a operação do servidor.

O arquivo resultante **`csr.pem`** contém sua chave pública, a assinatura digital de sua chave pública e os metadados que você inseriu.

4. Envie a CSR a uma CA. Geralmente, isso consiste em abrir seu arquivo de CSR em um editor de texto e copiar o conteúdo em um formulário da Web. No momento, você pode ser solicitado a fornecer um ou mais nomes alternativos de assunto (SANs) para serem colocados no certificado. Se **`www.example.com`** for o nome comum, **`example.com`** seria um bom SAN e vice-versa. Um visitante de seu site que digitar qualquer um desses nomes verá uma conexão livre de erros. Se o formulário web da CA permitir, inclua o nome comum na lista de SANs. Alguns o CAs incluem automaticamente.

   Depois que sua solicitação é aprovada, você recebe um novo certificado de host assinado pela CA. Você também pode receber uma instrução para fazer download de um arquivo de certificado

intermediário que contém os certificados adicionais necessários para concluir a cadeia de confiança da CA.

> **Note**
>
> Sua CA pode enviar a você arquivos em vários formatos com várias finalidades. Para este tutorial, você deve usar apenas um arquivo de certificado em formato PEM, que geralmente (mas nem sempre) é identificado por uma extensão de arquivo `.pem` ou `.crt`. Se você não tiver certeza sobre qual arquivo usar, abra os arquivos com um editor de texto e localize um que contenha um ou mais blocos com a linha a seguir.
>
> ```
> - - - - -BEGIN CERTIFICATE - - - - -
> ```
>
> O arquivo também deve terminar com a linha a seguir.
>
> ```
> - - - -END CERTIFICATE - - - - -
> ```
>
> Você também pode testar um arquivo na linha de comando da forma a seguir.
>
> ```
> [ec2-user certs]$ openssl x509 -in certificate.crt -text
> ```
>
> Verifique se as linhas aparecem no arquivo. Não use os arquivos que terminam com `.p7b`, `.p7c` ou extensões de arquivo semelhantes.

5. Coloque o novo certificado assinado pela CA e quaisquer certificados intermediários no diretório `/etc/pki/tls/certs`.

> **Note**
>
> Há várias maneiras de fazer o upload do seu novo certificado para sua EC2 instância, mas a maneira mais direta e informativa é abrir um editor de texto (por exemplo, vi, nano ou bloco de notas) no computador local e na instância e, em seguida, copiar e colar o conteúdo do arquivo entre eles. Você precisa de permissões root [sudo] ao realizar essas operações na EC2 instância. Dessa forma, você vê imediatamente se há algum problema de permissão ou de caminho. No entanto, tenha cuidado para não adicionar mais linhas ao copiar o conteúdo ou ao alterá-lo de alguma maneira.

De dentro do `/etc/pki/tls/certs` diretório, verifique se as configurações de propriedade, grupo e permissão do arquivo correspondem aos padrões AL2 023 altamente restritivos (proprietário = raiz, grupo = raiz, leitura/gravação somente para o proprietário). O exemplo a seguir mostra os comandos a serem usados.

```
[ec2-user certs]$ sudo chown root:root custom.crt
[ec2-user certs]$ sudo chmod 600 custom.crt
[ec2-user certs]$ ls -al custom.crt
```

Esses comandos devem produzir o resultado a seguir.

```
-rw------- root root custom.crt
```

As permissões para o arquivo de certificado intermediário são menos estritas (owner=root, group=root, proprietário pode gravar, grupo pode ler, mundo pode ler). O exemplo a seguir mostra os comandos a serem usados.

```
[ec2-user certs]$ sudo chown root:root intermediate.crt
[ec2-user certs]$ sudo chmod 644 intermediate.crt
[ec2-user certs]$ ls -al intermediate.crt
```

Esses comandos devem produzir o resultado a seguir.

```
-rw-r--r-- root root intermediate.crt
```

6. Coloque a chave privada que você usou para criar o CSR no diretório `/etc/pki/tls/private/`.

> **Note**
>
> Há várias maneiras de fazer upload da chave personalizada para a EC2 instância, mas a forma mais direta e informativa é abrir um editor de texto (por exemplo, vi, nano ou bloco de notas) no computador local e na instância e, em seguida, copiar e colar o conteúdo do arquivo entre eles. Você precisa de permissões root [sudo] ao realizar essas operações na EC2 instância. Dessa forma, você vê imediatamente se há algum

problema de permissão ou de caminho. No entanto, tenha cuidado para não adicionar mais linhas ao copiar o conteúdo ou ao alterá-lo de alguma maneira.

De dentro do `/etc/pki/tls/private` diretório, use os comandos a seguir para verificar se as configurações de propriedade, grupo e permissão do arquivo correspondem aos padrões AL2 023 altamente restritivos (owner=root, group=root, leitura/gravação somente para proprietário).

```
[ec2-user private]$ sudo chown root:root custom.key
[ec2-user private]$ sudo chmod 600 custom.key
[ec2-user private]$ ls -al custom.key
```

Esses comandos devem produzir o resultado a seguir.

```
-rw------- root root custom.key
```

7. Edite `/etc/httpd/conf.d/ssl.conf` para refletir seu novo certificado e arquivos de chave.

   a. Forneça o caminho e o nome do arquivo do certificado de host assinado por CA na diretiva `SSLCertificateFile` do Apache:

   ```
   SSLCertificateFile /etc/pki/tls/certs/custom.crt
   ```

   b. Se você receber um arquivo de certificado intermediário (`intermediate.crt` neste exemplo), forneça o caminho e o nome do arquivo usando a diretiva `SSLCACertificateFile` do Apache:

   ```
   SSLCACertificateFile /etc/pki/tls/certs/intermediate.crt
   ```

   > **Note**
   >
   > Alguns CAs combinam o certificado do host e os certificados intermediários em um único arquivo, tornando a `SSLCACertificateFile` diretiva desnecessária. Consulte as instruções fornecidas pela CA.

   c. Forneça o caminho e o nome do arquivo da chave privada (`custom.key` neste exemplo) na diretiva `SSLCertificateKeyFile` do Apache:

```
SSLCertificateKeyFile /etc/pki/tls/private/custom.key
```

8. Salve o `/etc/httpd/conf.d/ssl.conf` e reinicie o Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

9. Teste seu servidor inserindo seu nome de domínio em uma barra de URL do navegador com o prefixo `https://`. Seu navegador deve carregar a página de teste via HTTPS sem gerar erros.

Etapa 3: testar e intensificar a configuração de segurança

Depois que o SSL/TLS estiver operacional e exposto ao público, você precisará testar se ele é realmente seguro. É fácil fazer isso usando serviços online, como o Qualys SSL Labs que executa uma análise completa e gratuita de sua configuração de segurança. Com base nos resultados, você pode decidir intensificar a configuração de segurança padrão controlando quais protocolos você aceita, quais cifras você prefere e quais você exclui. Para obter mais informações, consulte como a Qualys formula suas pontuações.

> **Important**
>
> Os testes no mundo real são cruciais para a segurança do servidor. Pequenos erros de configuração podem resultar em rupturas de segurança sérias e em perda de dados. Como as práticas de segurança recomendadas são alteradas constantemente em resposta a pesquisas e a ameaças emergentes, auditorias periódicas da segurança são essenciais para uma boa administração do servidor.

No site Qualys SSL Labs, digite o nome do domínio totalmente qualificado de seu servidor no formato **`www.example.com`**. Depois de dois minutos, você recebe uma classificação (de A a F) para seu site e um detalhamento dos resultados. A tabela a seguir resume o relatório de um domínio com configurações idênticas à configuração padrão do Apache em AL2 023 e com um certificado padrão do Certbot.

Classificação geral	B
Certificado	100%

Suporte ao protocolo	95%
Troca de chaves	70%
Intensidade da cifra	90%

Embora a visão geral mostre que a configuração é mais sólida, o relatório detalhado sinaliza vários possíveis problemas, listados aqui em ordem de gravidade:

✗ A RC4 cifra é compatível com o uso de alguns navegadores mais antigos. Uma cifra é o núcleo matemático de um algoritmo de criptografia. RC4, uma cifra rápida usada para criptografar fluxos de dados TLS, é conhecida por ter várias fraquezas graves. A menos que você tenha boas razões para oferecer suporte a navegadores legados, você deve desabilitar isso.

✗ Versões antigas do TLS são compatíveis. A configuração é compatível com o TLS 1.0 (já obsoleto) e o TLS 1.1 (em um caminho para a reprovação). Apenas o TLS 1.2 é recomendado desde 2018.

✗ O sigilo de encaminhamento não é totalmente compatível. O sigilo encaminhado é um recurso de algoritmos que criptografam usando chaves de sessão temporárias (efêmeras) derivadas da chave privada. Na prática, isso significa que os atacantes não podem descriptografar dados HTTPS mesmo que tenham a chave privada de longo prazo de um servidor Web.

Para corrigir e preparar futuramente a configuração do TLS

1. Abra o arquivo de configuração `/etc/httpd/conf.d/ssl.conf` em um editor de texto e comente as seguintes linhas digitando “#” no início delas.

   ```
   #SSLProtocol all -SSLv3
   ```

2. Adicione a seguinte diretiva:

   ```
   #SSLProtocol all -SSLv3
   SSLProtocol -SSLv2 -SSLv3 -TLSv1 -TLSv1.1 +TLSv1.2
   ```

   Essa diretiva desabilita explicitamente as versões 2 e 3 do SSL, bem como as versões 1.0 e 1.1 do TLS. O servidor agora se recusa a aceitar conexões criptografadas com clientes que não estejam usando o TLS 1.2. A expressão detalhada na diretriz transmite mais claramente, para um leitor humano, para que o servidor está configurado.

> **Note**
>
> Desabilitar as versões 1.0 e 1.1 do TLS dessa forma bloqueia o acesso ao seu site de uma pequena porcentagem de navegadores da Web desatualizados.

Para modificar a lista de cifras permitidas

1. No arquivo de configuração `/etc/httpd/conf.d/ssl.conf`, localize a seção com a diretiva **`SSLCipherSuite`** e comente a linha existente ao inserir "#" no início dela.

   ```
   #SSLCipherSuite HIGH:MEDIUM:!aNULL:!MD5
   ```

2. Especifique conjuntos de criptografia explícitos e uma ordem de cifra que priorize o sigilo antecipado e evite cifras inseguras. A diretiva `SSLCipherSuite` usada aqui é baseada na saída do gerador de configuração SSL do Mozilla, que adapta uma configuração TLS ao software específico em execução no seu servidor. Primeiro, determine suas versões do Apache e do OpenSSL usando os comandos a seguir.

   ```
   [ec2-user ~]$ yum list installed | grep httpd

   [ec2-user ~]$ yum list installed | grep openssl
   ```

   Por exemplo, se a informação exibida for Apache 2.4.34 e OpenSSL 1.0.2, insira esses valores no gerador. Se você escolher o modelo de compatibilidade "moderno", isso criará uma diretiva `SSLCipherSuite` que impõe a segurança de forma agressiva, mas ainda funciona para a maioria dos navegadores. Se o software não oferecer suporte à configuração moderna, você poderá atualizá-lo ou escolher a configuração "intermediária".

   ```
   SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-
   ECDSA-CHACHA20-POLY1305:
   ECDHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-
   SHA256:
   ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-
   RSA-AES128-SHA256
   ```

As cifras selecionadas têm ECDHE em seus nomes, o que significa Elliptic Curve Diffie-Hellman Ephemeral (Curva elíptica de Diffie-Hellman efêmera). O termo ephemeral (efêmera) indica forward secrecy. Como subproduto, essas cifras não são compatíveis. RC4

Recomendamos que você use uma lista explícita de cifras em vez de confiar em padrões ou em diretrizes concisas cujo conteúdo não é visível.

Copie a diretiva gerada em `/etc/httpd/conf.d/ssl.conf`.

> **Note**
>
> Embora sejam mostradas em várias linhas aqui para facilitar a leitura, a diretriz deve estar em uma única linha quando copiada para `/etc/httpd/conf.d/ssl.conf` com apenas dois pontos (sem espaços) entre os nomes das cifras.

3. Por fim, remova o comentário da linha a seguir, excluindo o "#" no início dela.

```
#SSLHonorCipherOrder on
```

Essa diretriz força o servidor a preferir cifras de alta classificação incluindo (neste caso) aquelas que oferecem suporte a forward secrecy. Com essa diretriz ativada, o servidor tenta estabelecer uma conexão altamente segura antes de voltar a usar cifras permitidas com menos segurança.

Depois de concluir esses dois procedimentos, salve as alterações em `/etc/httpd/conf.d/ssl.conf` e reinicie o Apache.

Se você testar o domínio novamente no Qualys SSL Labs, verá que a RC4 vulnerabilidade e outros avisos desapareceram e que o resumo se parece com o seguinte.

Classificação geral	A
Certificado	100%
Suporte ao protocolo	100%
Troca de chaves	90%
Intensidade da cifra	90%

Cada atualização do OpenSSL apresenta novas cifras e retira o suporte às cifras antigas. Mantenha sua instância EC2 AL2 023 up-to-date, fique atento aos anúncios de segurança do OpenSSL e fique atento às denúncias de novas falhas de segurança na imprensa técnica.

Solução de problemas

- Meu servidor da web do Apache não inicia, a menos que eu digite uma senha.

  Esse é comportamento esperado se você tiver instalado uma chave privada de servidor criptografada e protegida por senha.

  Você pode remover a criptografia e a solicitação de senha da chave. Supondo que você tenha uma chave RSA criptografada privada chamada `custom.key` no diretório padrão e que a senha nela esteja **`abcde12345`**, execute os comandos a seguir na sua EC2 instância para gerar uma versão não criptografada da chave.

  ```
  [ec2-user ~]$ cd /etc/pki/tls/private/
  [ec2-user private]$ sudo cp custom.key custom.key.bak
  [ec2-user private]$ sudo openssl rsa -in custom.key -passin pass:abcde12345 -out
   custom.key.nocrypt
  [ec2-user private]$ sudo mv custom.key.nocrypt custom.key
  [ec2-user private]$ sudo chown root:root custom.key
  [ec2-user private]$ sudo chmod 600 custom.key
  [ec2-user private]$ sudo systemctl restart httpd
  ```

  O Apache agora deve iniciar sem solicitar uma senha a você.

- Obtenho erros ao executar sudo dnf install -y mod_ssl.

  Quando estiver instalando os pacotes necessários para SSL, você verá erros como os exibidos a seguir.

  ```
  Error: httpd24-tools conflicts with httpd-tools-2.2.34-1.16.amzn1.x86_64
  Error: httpd24 conflicts with httpd-2.2.34-1.16.amzn1.x86_64
  ```

  Isso normalmente significa que sua EC2 instância não está executando AL2 023. Este tutorial só é compatível com instâncias recém-criadas a partir de uma AL2 AMI 023 oficial.

Tutorial: Hospede um WordPress blog em AL2 023

Os procedimentos a seguir ajudarão você a instalar, configurar e proteger um WordPress blog na sua instância AL2 023. Este tutorial é uma boa introdução ao uso da Amazon EC2 , pois você tem controle total sobre um servidor web que hospeda seu WordPress blog, o que não é típico de um serviço de hospedagem tradicional.

Você é responsável para atualizar os pacotes de software e manter os patches de segurança para seu servidor. Para uma WordPress instalação mais automatizada que não exija interação direta com a configuração do servidor web, o AWS CloudFormation serviço fornece um WordPress modelo que também pode ajudar você a começar rapidamente. Para mais informações, consulte Get started (Conceitos básicos) no AWS CloudFormation User Guide (Guia do usuário do). Se você precisar de uma solução de alta disponibilidade com um banco de dados desacoplado, consulte Implantação de um WordPress site de alta disponibilidade no Guia do desenvolvedor.AWS Elastic Beanstalk

> **Important**
>
> Esses procedimentos são destinados ao uso com AL2 023. Para obter informações sobre outras distribuições, consulte a documentação específica. Muitas etapas deste tutorial não funcionam em instâncias Ubuntu. Para obter ajuda WordPress na instalação em uma instância do Ubuntu, consulte WordPressa documentação do Ubuntu. Você também pode usar CodeDeploypara realizar essa tarefa nos sistemas Amazon Linux, macOS ou Unix.

Tópicos

Pré-requisitos

É altamente recomendável que você associe um endereço IP elástico (EIP) à instância que você está usando para hospedar um WordPress blog. Isso impede que o endereço DNS público da sua instância mude e quebre sua instalação. Se você tiver um nome de domínio e quiser usá-lo para o blog, pode atualizar o registro DNS do nome de domínio para indicar ao seu endereço EIP (para

obter ajuda com isso, contate seu registrador de nome de domínio). Você pode ter um endereço EIP associado a uma instância em execução, gratuitamente. Para obter mais informações, consulte Endereços IP elásticos no Guia EC2 do usuário da Amazon. O tutorial Tutorial: instalar um servidor LAMP em AL2 023 apresenta etapas para configurar um grupo de segurança para permitir tráfego de HTTP e HTTPS, bem como várias etapas para garantir que as permissões de arquivos sejam definidas corretamente para seu servidor da Web. Para obter informações sobre como adicionar regras ao seu grupo de segurança, consulte Adicionar regras a um grupo de segurança.

Se você ainda não tiver um nome de domínio para seu blog, pode registrar um nome de domínio com o Route 53 e associar o endereço EIP de sua instância com seu nome de domínio. Para obter mais informações, consulte Registrar nomes de domínio usando o Amazon Route 53 no Guia do desenvolvedor do Amazon Route 53.

Instalar WordPress

Conecte-se à sua instância e baixe o pacote WordPress de instalação. Para obter mais informações sobre como se conectar à sua instância, consulte Conexão com AL2 203 instâncias.

1. Baixe e instale esses pacotes usando o comando a seguir.

```
dnf install wget php-mysqlnd httpd php-fpm php-mysqli mariadb105-server php-json
 php php-devel -y
```

2. Você pode notar um aviso exibido com verbiagem semelhante na saída (as versões podem variar com o tempo):

```
WARNING:
  A newer release of "Amazon Linux" is available.

  Available Versions:

dnf upgrade --releasever=2023.0.20230202

    Release notes:
     https://aws.amazon.com

  Version 2023.0.20230204:
    Run the following command to update to 2023.0.20230204:

      dnf upgrade --releasever=2023.0.20230204 ... etc
```

Como prática recomendada, recomendamos manter o sistema operacional o mais up-to-date possível, mas talvez você queira repetir cada versão para garantir que não haja conflitos em seu ambiente. Se a instalação dos pacotes anteriores anotados na etapa 1 falhar, talvez seja necessário atualizar para uma das versões mais recentes listadas e tentar novamente.

3. Baixe o pacote WordPress de instalação mais recente com o wget comando. O comando a seguir sempre deve baixar a versão mais recente.

```
[ec2-user ~]$ wget https://wordpress.org/latest.tar.gz
```

4. Descompacte e desarquive o pacote de instalação. A pasta de instalação é descompactada para uma pasta chamada `wordpress`.

```
[ec2-user ~]$ tar -xzf latest.tar.gz
```

Para criar um usuário de banco de dados e um banco de dados para sua WordPress instalação

Sua WordPress instalação precisa armazenar informações, como postagens de blog e comentários de usuários, em um banco de dados. Esse procedimento ajuda você a criar um banco de dados para seu blog e um usuário autorizado a ler e salvar as informações.

1. Inicie o servidor do banco de dados e da Web.

```
[ec2-user ~]$ sudo systemctl start mariadb httpd
```

2. Faça login no servidor do banco de dados como usuário `root`. Insira a senha de `root` do banco de dados quando solicitado; ela poderá ser diferente da sua senha do sistema de `root` ou poderá até estar vazia, se você não tiver protegido seu servidor do banco de dados.

   Se ainda não tiver protegido seu servidor do banco de dados, é muito importante que você faça isso. Para obter mais informações, consulte Etapa 3: Proteger o servidor do banco de dados (AL2023).

```
[ec2-user ~]$ mysql -u root -p
```

3. Crie um usuário e uma senha para seu banco de dados do MySQL. Sua WordPress instalação usa esses valores para se comunicar com seu banco de dados MySQL. Digite o comando a seguir, substituindo um nome de usuário e uma senha exclusivos.

```
CREATE USER 'wordpress-user'@'localhost' IDENTIFIED BY 'your_strong_password';
```

Crie uma senha forte para seu usuário. Não use o caractere de aspa única (') na sua senha, pois isso quebrará o comando anterior. Não reutilize uma senha existente e armazene essa senha em um lugar seguro.

4. Crie seu banco de dados. Dê ao seu banco de dados um nome descritivo e significativo, como `wordpress-db`.

> **Note**
>
> As marcas de pontuação que cercam o nome do banco de dados no comando abaixo são chamados backticks. A chave de backtick (`) costuma estar localizada acima da chave Tab de um teclado padrão. Backticks nem sempre são necessários, mas permitem que você use caracteres de outra forma ilegais, como hífens, no nome dos bancos de dados.

```
CREATE DATABASE `wordpress-db`;
```

5. Conceda privilégios totais do seu banco de dados ao WordPress usuário que você criou anteriormente.

```
GRANT ALL PRIVILEGES ON `wordpress-db`.* TO "wordpress-user"@"localhost";
```

6. Limpe os privilégios do banco de dados para receber todas as suas alterações.

```
FLUSH PRIVILEGES;
```

7. Saia do cliente `mysql`.

```
exit
```

Para criar e editar o arquivo wp-config.php

A pasta WordPress de instalação contém um exemplo de arquivo de configuração chamadowp-config-sample.php. Nesse procedimento, você copia esse arquivo e o edita para caber na sua configuração específica.

1. Copie o arquivo wp-config-sample.php para um arquivo chamado wp-config.php. Isso cria um novo arquivo de configuração e mantém o arquivo de exemplo original intacto como um backup.

```
[ec2-user ~]$ cp wordpress/wp-config-sample.php wordpress/wp-config.php
```

2. Edite o arquivo wp-config.php com seu editor de texto favorito (como o nano ou o vim) e insira os valores da instalação. Se você não tiver um editor de texto favorito, o nano é ideal para iniciantes.

```
[ec2-user ~]$ nano wordpress/wp-config.php
```

   a. Encontre a linha que define DB_NAME e altere database_name_here para o nome do banco de dados criado em Step 4 de Para criar um usuário de banco de dados e um banco de dados para sua WordPress instalação.

```
define('DB_NAME', 'wordpress-db');
```

   b. Encontre a linha que define DB_USER e altere username_here para o usuário do banco de dados que você criou Step 3 de Para criar um usuário de banco de dados e um banco de dados para sua WordPress instalação.

```
define('DB_USER', 'wordpress-user');
```

   c. Encontre a linha que define DB_PASSWORD e altere password_here para a senha mais forte que você criou em Step 3 de Para criar um usuário de banco de dados e um banco de dados para sua WordPress instalação.

```
define('DB_PASSWORD', 'your_strong_password');
```

   d. Encontre a seção chamada Authentication Unique Keys and Salts. Esses SALT valores KEY e esses fornecem uma camada de criptografia aos cookies do navegador que WordPress os usuários armazenam em suas máquinas locais. Basicamente, adicionar

valores longos e aleatórios aqui deixa seu site mais seguro. Visite https://api.wordpress. org/secret-key/1.1/salt/para gerar aleatoriamente um conjunto de valores-chave que você pode copiar e colar em seu `wp-config.php` arquivo. Para colar texto em um terminal do PuTTY, coloque o cursor onde deseja colar texto e clique com o botão direito do mouse dentro do terminal do PuTTY.

Para obter mais informações sobre chaves de segurança, acesse https://wordpress. org/support/article/editing-wp-config-php/#security -keys.

> **Note**
>
> Os valores abaixo são somente para fins de exemplo; não use esses valores para a instalação.

```
define('AUTH_KEY',         ' #U$$+[RXN8:b^-L 0(WU_+ c+WFkI~c]o]-bHw+)/
Aj[wTwSiZ<Qb[mghEXcRh-');
define('SECURE_AUTH_KEY',  'Zsz._P=l/|y.Lq)XjlkwS1y5NJ76E6EJ.AV0pCKZZB,*~*r ?
6OP$eJT@;+(ndLg');
define('LOGGED_IN_KEY',    'ju}qwre3V*+8f_zOWf?{LlGsQ]Ye@2Jh^,8x>)Y |;(^[Iw]Pi
+LG#A4R?7N`YB3');
define('NONCE_KEY',        'P(g62HeZxEes|LnI^i=H,[XwK9I&[2s|:?0N}VJM%?;v2v]v+;
+^9eXUahg@::Cj');
define('AUTH_SALT',        'C$DpB4Hj[JK:?{ql`sRVa:{:7yShy(9A@5wg+`JJVb1fk%_-
Bx*M4(qc[Qg%JT!h');
define('SECURE_AUTH_SALT', 'd!uRu#}+q#{f$Z?Z9uFPG.${+S{n~1M&%@~gL>U>NV<zpD-@2-
Es7Q1O-bp28EKv');
define('LOGGED_IN_SALT',   ';j{00P*owZf)kVD+FVLn-~ >.|Y%Ug4#I^*LVd9QeZ^&XmK|
e(76miC+&W&+^0P/');
define('NONCE_SALT',       '-97r*V/cgxLmp?Zy4zUU4r99QQ_rGs2LTd%P;|
_e1tS)8_B/,.6[=UK<J_y9?JWG');
```

e. Salve o arquivo e saia do seu editor de texto.

Para instalar seus WordPress arquivos na raiz do documento Apache

- Agora que você descompactou a pasta de instalação, criou um banco de dados e um usuário MySQL e personalizou o arquivo de WordPress configuração, você está pronto para copiar os arquivos de instalação para a raiz do documento do servidor web para poder executar o script de instalação que conclui a instalação. A localização desses arquivos depende se

você deseja que seu WordPress blog esteja disponível na raiz real do seu servidor web (por exemplo,*my.public.dns.amazonaws.com*) ou em um subdiretório ou pasta abaixo da raiz (por exemplo,*my.public.dns.amazonaws.com/blog*).

- Se você quiser WordPress executar na raiz do seu documento, copie o conteúdo do diretório de instalação do wordpress (mas não o diretório em si) da seguinte forma:

```
[ec2-user ~]$ cp -r wordpress/* /var/www/html/
```

- Se você quiser WordPress executar em um diretório alternativo na raiz do documento, primeiro crie esse diretório e, em seguida, copie os arquivos nele. Neste exemplo, WordPress será executado a partir do diretório`blog`:

```
[ec2-user ~]$ mkdir /var/www/html/blog
[ec2-user ~]$ cp -r wordpress/* /var/www/html/blog/
```

> **Important**
>
> Para fins de segurança, se você não estiver segundo para o procedimento seguinte imediatamente, pare o Apache Web Server (`httpd`) agora. Depois de mover sua instalação para a raiz do documento Apache, o script de WordPress instalação fica desprotegido e um invasor pode obter acesso ao seu blog se o servidor web Apache estiver em execução. Para interromper o servidor web Apache, insira o comando sudo service httpd stop. Se você estiver passando para o procedimento seguinte, não precisa parar o Apache Web Server.

Para permitir o uso WordPress de links permanentes

WordPress os permalinks precisam usar `.htaccess` arquivos Apache para funcionarem corretamente, mas isso não está habilitado por padrão no Amazon Linux. Use o procedimento a seguir para permitir todas as substituições na raiz de documentos do Apache.

1. Abra o arquivo `httpd.conf` com seu editor de texto de preferência (como nano ou vim). Se você não tiver um editor de texto favorito, o `nano` é ideal para iniciantes.

```
[ec2-user ~]$ sudo vim /etc/httpd/conf/httpd.conf
```

2. Encontre a seção que começa com `<Directory "/var/www/html">`.

```
<Directory "/var/www/html">
    #
    # Possible values for the Options directive are "None", "All",
    # or any combination of:
    #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
    #
    # Note that "MultiViews" must be named *explicitly* --- "Options All"
    # doesn't give it to you.
    #
    # The Options directive is both complicated and important.  Please see
    # http://httpd.apache.org/docs/2.4/mod/core.html#options
    # for more information.
    #
    Options Indexes FollowSymLinks

    #
    # AllowOverride controls what directives may be placed in .htaccess files.
    # It can be "All", "None", or any combination of the keywords:
    #   Options FileInfo AuthConfig Limit
    #
    AllowOverride None

    #
    # Controls who can get stuff from this server.
    #
    Require all granted
</Directory>
```

3. Altere a linha `AllowOverride None` na seção acima para `AllowOverride All`.

> **Note**
>
> Há múltiplas linhas `AllowOverride` nesse arquivo; altere a linha na seção `<Directory "/var/www/html">`.

```
AllowOverride All
```

4. Salve o arquivo e saia do seu editor de texto.

Para instalar a biblioteca de desenhos gráficos PHP em AL2 023

A biblioteca de desenhos gráficos para PHP permite modificar imagens. Instale esta biblioteca caso você precise cortar a imagem do cabeçalho do blog. A versão phpMyAdmin que você instala pode exigir uma versão mínima específica dessa biblioteca (por exemplo, versão 8.1).

Use o comando a seguir para instalar a biblioteca de desenhos gráficos PHP em AL2 023. Por exemplo, se você instalou o php8.1 da origem como parte da instalação da pilha LAMP, este comando instalará a versão 8.1 da biblioteca de desenhos gráficos PHP.

```
[ec2-user ~]$ sudo dnf install php-gd
```

Para verificar a versão instalada, use o seguinte comando:

```
[ec2-user ~]$ sudo dnf list installed | grep php-gd
```

A seguir está um exemplo de saída:

```
php-gd.x86_64                    8.1.30-1.amzn2           @amazonlinux
```

Para instalar a biblioteca de desenhos gráficos PHP no Amazon Linux AMI

A biblioteca de desenhos gráficos para PHP permite modificar imagens. Instale esta biblioteca caso você precise cortar a imagem do cabeçalho do blog. A versão phpMyAdmin que você instala pode exigir uma versão mínima específica dessa biblioteca (por exemplo, versão 8.1).

Para verificar quais versões estão disponíveis, use o seguinte comando:

```
[ec2-user ~]$ dnf list | grep php
```

A seguir são mostradas linhas de exemplo da saída para a biblioteca de desenhos gráficos PHP (versão 8.1):

```
php8.1.aarch64                                                  8.1.7-1.amzn2023.0.1
                      @amazonlinux
php8.1-cli.aarch64                                              8.1.7-1.amzn2023.0.1
                      @amazonlinux
php8.1-common.aarch64                                           8.1.7-1.amzn2023.0.1
                      @amazonlinux
```

```
php8.1-devel.aarch64                                    8.1.7-1.amzn2023.0.1
                        @amazonlinux
php8.1-fpm.aarch64                                      8.1.7-1.amzn2023.0.1
                        @amazonlinux
php8.1-gd.aarch64                                       8.1.7-1.amzn2023.0.1
                        @amazonlinux
```

Use o comando a seguir para instalar uma versão específica da biblioteca de desenhos gráficos PHP (por exemplo, php8.1) no Amazon Linux AMI:

```
[ec2-user ~]$ sudo dnf install -y php8.1-gd
```

Para corrigir as permissões de arquivos para o Apache Web Server

Alguns dos recursos disponíveis WordPress exigem acesso de gravação à raiz do documento Apache (como o upload de mídia pelas telas de administração). Se você não tiver feito isso, aplique as associações e permissões de grupo a seguir (conforme descrito em mais detalhes no tutorial do servidor web LAMP).

1. Conceda a propriedade do arquivo de `/var/www` e seu conteúdo para o usuário `apache`.

   ```
   [ec2-user ~]$ sudo chown -R apache /var/www
   ```

2. Conceda a propriedade do grupo do `/var/www` e seu conteúdo para o grupo do `apache`.

   ```
   [ec2-user ~]$ sudo chgrp -R apache /var/www
   ```

3. Altere as permissões do diretório do `/var/www` e de seus subdiretórios para adicionar permissões de gravação do grupo e definir o ID do grupo em subdiretórios futuros.

   ```
   [ec2-user ~]$ sudo chmod 2775 /var/www
   [ec2-user ~]$ find /var/www -type d -exec sudo chmod 2775 {} \;
   ```

4. Altere recursivamente as permissões de arquivo de `/var/www` e de seus subdiretórios.

   ```
   [ec2-user ~]$ find /var/www -type f -exec sudo chmod 0644 {} \;
   ```

> **Note**
>
> Se você também pretende usar WordPress como servidor FTP, precisará de configurações de grupo mais permissivas aqui. Revise as etapas recomendadas e as configurações de segurança WordPress para fazer isso.

5. Reinicie o Apache Web Server para pegar o grupo e as permissões novos.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

Para executar o script WordPress de instalação com AL2 023

Você está pronto para instalar WordPress. Os comandos usados por você dependem do sistema operacional. Os comandos neste procedimento são para uso com AL2 023. Use o procedimento que segue este com AL2 023 AMI.

1. Use o comando systemctl para garantir que `httpd` e os serviços do banco de dados sejam iniciados a cada inicialização do sistema.

```
[ec2-user ~]$ sudo systemctl enable httpd && sudo systemctl enable mariadb
```

2. Verifique se o servidor do banco de dados está em execução.

```
[ec2-user ~]$ sudo systemctl status mariadb
```

   Se o serviço do banco de dados não está em execução, inicie-o.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

3. Verifique se o Apache Web Server (`httpd`) está sendo executado.

```
[ec2-user ~]$ sudo systemctl status httpd
```

   Se o serviço `httpd` não estiver sendo executado, inicie-o.

```
[ec2-user ~]$ sudo systemctl start httpd
```

4. Em um navegador da Web, digite a URL do seu WordPress blog (o endereço DNS público da sua instância ou o endereço seguido pela `blog` pasta). Você deve ver o script WordPress de instalação. Forneça as informações exigidas pela WordPress instalação. Escolha Instalar WordPress para concluir a instalação. Para obter mais informações, consulte Etapa 5: Executar o script de instalação no WordPress site.

Para executar o script WordPress de instalação com AL2 023 AMI

1. Use o comando chkconfig para garantir que `httpd` e os serviços do banco de dados sejam iniciados a cada inicialização do sistema.

```
[ec2-user ~]$ sudo chkconfig httpd on && sudo chkconfig mariadb on
```

2. Verifique se o servidor do banco de dados está em execução.

```
[ec2-user ~]$ sudo service mariadb status
```

   Se o serviço do banco de dados não está em execução, inicie-o.

```
[ec2-user ~]$ sudo service mariadb start
```

3. Verifique se o Apache Web Server (`httpd`) está sendo executado.

```
[ec2-user ~]$ sudo service httpd status
```

   Se o serviço `httpd` não estiver sendo executado, inicie-o.

```
[ec2-user ~]$ sudo service httpd start
```

4. Em um navegador da Web, digite a URL do seu WordPress blog (o endereço DNS público da sua instância ou o endereço seguido pela `blog` pasta). Você deve ver o script WordPress de instalação. Forneça as informações exigidas pela WordPress instalação. Escolha Instalar WordPress para concluir a instalação. Para obter mais informações, consulte Etapa 5: Executar o script de instalação no WordPress site.

Próximas etapas

Depois de testar seu WordPress blog, considere atualizar sua configuração.

Usar um nome de domínio personalizado

Se você tiver um nome de domínio associado ao endereço EIP da sua EC2 instância, poderá configurar seu blog para usar esse nome em vez do endereço DNS EC2 público. Para obter mais informações, consulte Alterando a URL do WordPress site no site.

Configurar seu blog

Você pode configurar seu blog para usar diferentes temas e plug-ins e oferecer uma experiência mais personalizada para seus leitores. Contudo, às vezes o processo de instalação pode dar errado, fazendo com que você perca o blog inteiro. Recomendamos veementemente que você crie um backup da imagem de máquina da Amazon (AMI) de sua instância antes de tentar instalar quaisquer temas ou plug-ins, de forma que consiga restaurar o blog se algo der errado durante a instalação. Para obter mais informações, consulte Crie sua própria AMI no Guia EC2 do usuário da Amazon.

Aumentar a capacidade

Se seu WordPress blog se tornar popular e você precisar de mais capacidade computacional ou armazenamento, considere as seguintes etapas:

- Expanda o espaço de armazenamento na sua instância. Para obter mais informações, consulte Amazon EBS Elastic Volumes.
- Mova o banco de dados MySQL para o Amazon RDS para aproveitar a capacidade de dimensionamento que o serviço oferece.

Melhore a performance de rede do tráfego da Internet

Se você espera que seu blog gere tráfego de usuários localizados em todo o mundo, considere o AWS Global Accelerator. O Global Accelerator ajuda você a obter menor latência melhorando o desempenho do tráfego da Internet entre os dispositivos cliente de seus usuários e seu WordPress aplicativo em execução. AWS O Global Accelerator usa a rede AWS global para direcionar o tráfego para um endpoint de aplicativo saudável na AWS região mais próxima do cliente.

Saiba mais sobre WordPress

Os links a seguir contêm mais informações sobre WordPress.

- Para obter informações sobre WordPress, consulte a documentação de ajuda do WordPress Codex no Codex.

- Para obter mais informações sobre como solucionar problemas de instalação, acesse Problemas comuns de instalação.
- Para obter informações sobre como tornar seu WordPress blog mais seguro, acesse Fortalecimento WordPress.
- Para obter informações sobre como manter seu WordPress blog up-to-date, acesse Atualizar WordPress.

Ajuda! Meu nome DNS público mudou e agora meu blog não está funcionando

Sua WordPress instalação é configurada automaticamente usando o endereço DNS público da sua EC2 instância. Se você parar e reiniciar a instância, o endereço DNS público mudará (a menos que esteja associado a um endereço IP elástico) e seu blog não funcionará mais porque faz referência a recursos em um endereço que não existe mais (ou está atribuído a outra EC2 instância). Uma descrição mais detalhada do problema e várias soluções possíveis estão descritas em https://wordpress. org/support/article/changing-the-site-url/.

Se isso aconteceu com sua WordPress instalação, talvez você consiga recuperar seu blog com o procedimento abaixo, que usa a interface de linha de wp-cli comando para WordPress.

Para alterar o URL WordPress do seu site com o wp-cli

1. Conecte-se à sua EC2 instância com SSH.
2. Anote o URL do site antigo e do site novo para sua instância. O URL antigo do site provavelmente é o nome DNS público da sua EC2 instância quando você instalou WordPress. O novo URL do site é o nome DNS público atual da sua EC2 instância. Se você não tiver certeza da URL do site antigo, pode usar o curl para encontrá-la com o seguinte comando.

   ```
   [ec2-user ~]$ curl localhost | grep wp-content
   ```

   Você deve visualizar referências ao nome DNS público antigo na saída, que terá a seguinte aparência (URL do site antigo em vermelho):

   ```
   <script type='text/javascript' src='http://ec2-52-8-139-223.us-
   west-1.compute.amazonaws.com/wp-content/themes/twentyfifteen/js/functions.js?
   ver=20150330'></script>
   ```

3. Faça download do wp-cli com o seguinte comando.

```
[ec2-user ~]$ curl -O https://raw.githubusercontent.com/wp-cli/builds/gh-pages/
phar/wp-cli.phar
```

4. Pesquise e substitua o URL antigo do site em sua WordPress instalação pelo comando a seguir. Substitua sua EC2 instância URLs pelo site antigo e pelo novo e o caminho para sua WordPress instalação (geralmente `/var/www/html` ou`/var/www/html/blog`).

```
[ec2-user ~]$ php wp-cli.phar search-replace 'old_site_url' 'new_site_url' --path=/
path/to/wordpress/installation --skip-columns=guid
```

5. Em um navegador da Web, insira a nova URL do seu WordPress blog para verificar se o site está funcionando corretamente novamente. Se não estiver, consulte Alteração da URL do site e Problemas comuns de instalação para obter mais informações.

Tutorial: Transição do Redis 6 para o Valkey em 023 AL2

A documentação a seguir descreve os principais aspectos da transição do Redis 6 para o Valkey em AL2 023.

Cronograma de suporte para Redis 6

O Redis 6 chega ao fim da vida útil (EOL) em 31 de agosto de 2025. Após essa data, o Redis 6 não receberá mais atualizações ou patches de segurança do projeto Redis. É altamente recomendável que os usuários migrem para o Valkey antes de agosto de 2025 para garantir suporte contínuo e atualizações de segurança.

Para obter mais informações sobre os cronogramas de suporte da versão do Redis, consulte a documentação do Redis End-Of-Life Schedule.

Introdução ao Valkey

O Valkey é um fork de código aberto do Redis 7, mantido pela The Linux Foundation. É totalmente compatível com as versões 2.x a 7.2.x do Redis Open Source Software (OSS). A Valkey mantém a API e a funcionalidade conhecidas do Redis, ao mesmo tempo que oferece vários aprimoramentos:

- Desempenho aprimorado por meio de multisegmentação.
- Maior eficiência de memória, especialmente no modo cluster.

- Replicação de canal duplo para melhor consistência dos dados.

Plano e cronograma de migração

É altamente recomendável que os usuários migrem do Redis 6 para o Valkey antes de 31 de agosto de 2025, quando o Redis 6 chegar ao fim da vida útil (EOL). Essa migração requer intervenção manual e não é automática.

O Amazon Linux recomenda essa migração para garantir atualizações contínuas de funcionalidade, suporte e segurança para seus aplicativos dependentes do Redis.

Opções e etapas de migração

Propomos três caminhos de migração para o Valkey com base em seus requisitos de implantação e necessidades operacionais.

Opção 1: instalação de nova instância

Para novas implantações ou quando a migração de dados não é necessária:

1. Instale o Valkey:

```
[ec2-user ~]$ sudo dnf install valkey
```

2. Inicie o Vale:

```
[ec2-user ~]$ sudo systemctl start valkey
```

3. (Opcional) Ative o Valkey na inicialização:

```
[ec2-user ~]$ sudo systemctl enable valkey
```

4. Verifique a instalação:

```
[ec2-user ~]$ valkey-cli info server
[ec2-user ~]$ valkey-cli ping
```

Opção 2: substituição no local

Para instâncias existentes em que a persistência de dados não é necessária:

1. Pare o Redis 6:

```
[ec2-user ~]$ sudo systemctl stop redis6
```

2. Instale o Valkey:

```
[ec2-user ~]$ sudo dnf install valkey
```

3. (Opcional) Use a configuração do Redis 6 no Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/redis6.conf /etc/valkey/valkey.conf
[ec2-user ~]$ sudo cp /etc/valkey/valkey.conf /etc/valkey/valkey.conf.backup
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/valkey.conf
[ec2-user ~]$ sudo sed -i 's|^dir\s.*|dir /var/lib/valkey|g' /etc/valkey/
valkey.conf
```

4. (Opcional) Use o arquivo de configuração sentinel do Redis 6 no Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/sentinel.conf /etc/valkey/sentinel.conf
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/sentinel.conf
```

5. Inicie o Vale:

```
[ec2-user ~]$ sudo systemctl start valkey
```

6. (Opcional) Ative o Valkey na inicialização:

```
[ec2-user ~]$ sudo systemctl enable valkey
```

7. Verifique a instalação do Valkey:

```
[ec2-user ~]$ valkey-cli info server
[ec2-user ~]$ valkey-cli ping
```

8. Remova o Redis 6:

```
[ec2-user ~]$ sudo dnf remove redis6
```

Opção 3: migração de dados

Essa opção permite que você execute o Redis 6 e o Valkey simultaneamente.

1. Instale o Valkey sem remover o Redis 6:

```
[ec2-user ~]$ sudo dnf install valkey
```

2. (Opcional) Use a configuração do Redis 6 no Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/redis6.conf /etc/valkey/valkey.conf
[ec2-user ~]$ sudo cp /etc/valkey/valkey.conf /etc/valkey/valkey.conf.backup
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/valkey.conf
[ec2-user ~]$ sudo sed -i 's|^dir\s.*|dir /var/lib/valkey|g' /etc/valkey/
valkey.conf
```

3. (Opcional) Use o arquivo de configuração sentinel do Redis 6 no Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/sentinel.conf /etc/valkey/sentinel.conf
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/sentinel.conf
```

4. Modifique a configuração do Valkey:

   Edite `/etc/valkey/valkey.conf` e defina a diretiva 'port' com um valor diferente (por exemplo, 6380) para evitar conflitos com o Redis 6.

5. Inicie o Vale:

```
[ec2-user ~]$ sudo systemctl start valkey
```

6. (Opcional) Ative o Valkey na inicialização:

```
[ec2-user ~]$ sudo systemctl enable valkey
```

7. Verifique a instalação do Valkey:

```
[ec2-user ~]$ valkey-cli -p port info server
[ec2-user ~]$ valkey-cli -p port ping
```

> **Note**
>
> *port*Substitua pelo número da porta configurada.

8. Migre dados:

Agora você pode migrar dados do Redis 6 para o Valkey usando métodos de replicação ou transferência manual de dados.

9. Atualize as configurações do aplicativo:

   Atualize gradualmente seus aplicativos para usar a porta Valkey.

10. Remova o Redis 6:

    Depois que todos os dados e aplicativos tiverem sido migrados, você poderá interromper e remover o Redis 6.

```
[ec2-user ~]$ sudo systemctl stop redis6
[ec2-user ~]$ sudo dnf remove redis6
```

> **Note**
>
> É altamente recomendável validar o processo de migração em um ambiente de teste antes de implementar mudanças nos sistemas de produção.

Tópicos relacionados

Para obter mais informações sobre Valkey:

- Vale: https://valkey.io/
- Migração do Vale: https://valkey. io/topics/migration/

Tutorial: Instale o ambiente de trabalho GNOME em 023 AL2

O ambiente de trabalho GNOME está disponível como uma interface gráfica de usuário opcional para AL2 0.23 a partir da versão 2023.7 ou posterior.

Os procedimentos a seguir ajudam você a instalar o ambiente de trabalho GNOME na sua instância AL2 023. Você pode usar essa interface gráfica para interagir com seu sistema Linux usando um ambiente de desktop familiar em vez de apenas a interface de linha de comando.

Conteúdo

Pré-requisitos

- O ambiente de desktop requer pelo menos 2,4 GB de memória. Portanto, uma instância do tipo `t2.medium` ou melhor é recomendada para garantir um desempenho adequado. Exemplos de tipos de instância com memória insuficiente incluem `t2.nano` `t2.micro`, `t2.small` e. Essa restrição também se aplica a `t4` instâncias desse tamanho `t3` e a qualquer outro tipo de instância que não atenda aos requisitos de memória.
- Este tutorial pressupõe que você já tenha executado uma instância usando a versão AL2 023 executando a versão 2023.7 ou posterior. Para obter mais informações, consulte AL2023 na Amazon EC2 as Atualizando AL2 023 páginas e.

Instalação

- Instale o ambiente de trabalho GNOME e os pacotes relacionados.

```
[ec2-user ~]$ sudo dnf groupinstall "Desktop" -y
```

> **Note**
>
> Para acessar o ambiente de desktop gráfico, você precisará instalar e configurar software adicional, como Amazon DCV ou VNC. Essas ferramentas permitem que você se conecte e interaja com a interface gráfica do usuário na rede.

Tópicos relacionados

Para obter mais informações sobre o ambiente de desktop gráfico, consulte a seguinte documentação:

- O que é o Amazon DCV? no Guia do administrador do Amazon DCV
- Tutorial: Configurar o servidor TigerVNC em 023 AL2

Tutorial: Configurar o servidor TigerVNC em 023 AL2

Os procedimentos a seguir ajudam você a configurar o servidor VNC na sua instância AL2 023. O VNC permite que você acesse e interaja remotamente com o ambiente gráfico da área de trabalho por meio de uma conexão de rede segura.

Conteúdo

Pré-requisitos

- Este tutorial pressupõe que você já tenha instalado o ambiente de desktop GNOME na sua instância AL2 023. Para obter mais informações, consulte a página do Tutorial: Instale o ambiente de trabalho GNOME em 023 AL2.
- Este tutorial usa o encaminhamento de porta SSH para acessar o servidor VNC. Para obter mais informações sobre como configurar seu key pair, consulte Connect to your Linux instance using SSH no Amazon EC2 User Guide.
- O procedimento a seguir não orienta você no processo de instalação de um cliente VNC. Você deve ter um cliente VNC instalado em sua máquina local para poder se conectar e interagir com o ambiente de desktop.

Etapa 1: Instalação

1. Conecte-se à sua instância. Para obter mais informações, consulte Conexão com AL2 203 instâncias.
2. Instale o pacote do servidor TigerVNC para 023. AL2

A `-y` opção instala o pacote sem pedir confirmação. Se quiser examinar o pacote antes da instalação, você pode omitir essa opção.

```
[ec2-user ~]$ sudo dnf install -y tigervnc-server
```

Etapa 2: Configuração

1. Verifique se o usuário configurou uma senha VNC.

```
[ec2-user ~]$ vncpasswd
```

2. Atribua um número de exibição ao usuário.

```
[ec2-user ~]$ sudo vi /etc/tigervnc/vncserver.users
```

Adicione a seguinte configuração:

```
:1=ec2-user
```

> **Note**
>
> Você pode atribuir qualquer número de exibição ao usuário. Estamos usando display `:1` para este exemplo.

3. Edite o arquivo de configuração do servidor VNC.

```
[ec2-user ~]$ sudo vi /etc/tigervnc/vncserver-config-defaults
```

Adicione a seguinte configuração:

```
session=gnome
securitytypes=vncauth,tlsvnc
geometry=1920x1080
localhost
alwaysshared
```

> **Note**
>
> Você pode alterar a resolução da tela usando o `geometry` parâmetro. Estamos usando `1920x1080` para fins deste exemplo.

4. Inicie o servidor VNC. Esse processo precisa ser repetido toda vez que você reinicia sua instância. Se você quiser automatizar o processo de inicialização desse serviço, consulte a seção opcional abaixo.

```
[ec2-user ~]$ sudo systemctl start vncserver@:1
```

> **Important**
>
> Ao iniciar o `vncserver` serviço, a peça após a `@` deve corresponder ao número de exibição definido para o usuário no `/etc/tigervnc/vncserver.users` arquivo.

Depois de executar esta etapa, você pode criar o túnel SSH a partir de sua máquina local e se conectar usando seu cliente VNC.

Etapa 3: Conecte-se usando um cliente VNC

O servidor VNC expõe um soquete TCP para conexões de clientes. Embora você possa expor a porta VNC diretamente por meio do seu grupo de segurança, este tutorial demonstra o uso do tunelamento SSH como uma abordagem mais segura, criptografando a conexão entre sua máquina local e a instância. EC2 Depois de conectado pelo túnel, você se autenticará no servidor VNC usando a senha que você configurou na etapa anterior. Para obter mais informações sobre grupos de segurança, consulte Alterar os grupos de segurança da sua EC2 instância da Amazon no Guia EC2 do usuário da Amazon.

1. Crie um túnel SSH a partir da sua máquina local.

```
$ ssh -i <keypair> -L 5901:localhost:5901 ec2-user@<address>
```

> **Note**
>
> `<keypair>`Substitua pelo caminho para sua chave SSH e `<address>` pelo IP público ou nome DNS da sua instância. A porta muda com base no número de exibição usado para iniciar `vncserver` o. Por exemplo, o monitor `:1` usa a porta`5901`, o monitor `:2` usa a porta `5902` etc.

2. Use seu cliente VNC para se conectar à `localhost:5901` ou `127.0.0.1:5901` com a senha VNC definida anteriormente.

> **Important**
>
> Mantenha o túnel SSH aberto ao usar o VNC. Se o túnel SSH não estiver aberto, você não poderá usar seu cliente VNC para visualizar e interagir com o ambiente de desktop.

(Opcional) Inicie o serviço na inicialização

Se você planeja usar o VNC regularmente, talvez queira configurar o servidor VNC para iniciar automaticamente quando sua instância for inicializada. Isso elimina a necessidade de iniciar manualmente o servidor VNC sempre que você reinicia sua instância. Essa configuração garante que seu ambiente gráfico de desktop esteja pronto e acessível assim que sua instância concluir o processo de inicialização.

- Configure o serviço para iniciar na inicialização.

```
[ec2-user ~]$ sudo systemctl enable vncserver@:1
```

> **Important**
>
> Ao ativar o `vncserver` serviço, a peça após o `@` deve corresponder ao número de exibição definido para o usuário no `/etc/tigervnc/vncserver.users` arquivo. Além disso, você pode transmitir o `--now` argumento after `enable` para iniciar o serviço imediatamente.

Depois de realizar essa etapa, você não precisará mais iniciar `vncserver` toda vez que reinicializar sua instância.

(Opcional) Desative a tela de bloqueio inativa

- Defina o atraso de inatividade para zero para desativar a tela de bloqueio quando o usuário estiver inativo por um longo período de tempo.

```
[ec2-user ~]$ gsettings set org.gnome.desktop.session idle-delay 0
```

Tópicos relacionados

Para obter mais informações sobre o ambiente de desktop gráfico, consulte a seguinte documentação:

- Tutorial: Instale o ambiente de trabalho GNOME em 023 AL2
- O que é o Amazon DCV? no Guia do administrador do Amazon DCV

Usando o Amazon Linux 2023 fora da Amazon EC2

As imagens do contêiner Amazon Linux 2023 podem ser executadas em ambientes de tempo de execução de contêineres compatíveis. Para obter mais informações sobre como usar o Amazon Linux 2023 dentro de um contêiner, consulte AL2023 em containers.

O Amazon Linux 2023 (AL2023) também pode ser executado como um convidado virtualizado, além de ser executado diretamente na Amazon. EC2 Atualmente, existem KVM Imagens (`qcow2`), VMware (`OVA`) e Hyper-V (`vhdx`) disponíveis.

> **Note**
>
> A configuração das imagens do Amazon Linux 2023 é diferente da do Amazon Linux 2. Se você estiver executando o Amazon Linux 2 como uma máquina virtual no local, precisará adaptar sua configuração para ser compatível com AL2 023.

Baixe imagens do Amazon Linux 2023 para uso com KVM e VMware Hyper-V

As imagens de disco do Amazon Linux 2023 para uso com KVM e Hyper-V podem ser VMware baixadas em cdn.amazonlinux.com.

Configurações suportadas do Amazon Linux 2023 para uso em ambientes virtualizados não pertencentes à Amazon EC2

Esta seção aborda os requisitos para executar o Amazon Linux 2023 em ambientes EC2 virtualizados que não sejam da Amazon, como no KVM ou no VMware Hyper-V.

A base AL2023 requisitos do sistema se aplica a todos os ambientes EC2 virtualizados que não são da Amazon. Uma lista dos modelos de dispositivos compatíveis é detalhada para cada ambiente de hipervisor nos tópicos a seguir.

O KVM e o Hyper-V oferecem muitas opções de configuração, e é preciso tomar cuidado para configurá-las de acordo com suas necessidades de segurança, desempenho e confiabilidade. VMware Para obter mais informações, consulte a documentação fornecida pelo seu hipervisor.

Tópicos

Requisitos para executar o AL2 023 no KVM

Esta seção descreve os requisitos para executar o AL2 023 no KVM. As imagens KVM de AL2 023 estão disponíveis para ambas `aarch64` as arquiteturas. `x86-64` Esses requisitos são adicionais à base AL2023 requisitos do sistema para as imagens KVM.

Tópicos

Requisitos de host KVM para executar AL2 023 no KVM

As imagens KVM estão atualmente qualificadas em um host executando o Ubuntu 22.04.3 LTS com a versão `qemu` 6.2+dfsg-2ubuntu6.15, fornecido por esta versão do Ubuntu, usando um tipo `q35` de máquina para `x86-64` e um tipo `virt` de máquina para`aarch64`.

Suporte de dispositivo para AL2 023 no KVM

Os modelos de **`qemu`** dispositivos testados para uso com AL2 023 imagens KVM (ambas **`aarch64`** e**`x86-64`**) são:

- `virtio-blk` (dispositivo de bloco `virtio`)
- `virtio-scsi` (`virtio` SCSI controlador (com dispositivo de disco)
- `virtio-net` (dispositivo `virtio` de rede)
- `ahci` (para uso com a unidade de CD-ROM virtual)
- `usb-storage` (sobre `xhci`)

Modelos de **`qemu`** dispositivos adicionais habilitados na qualificação de imagem AL2 023 KVM, mas não muito exercitados, são:

- `VGA (qemu VGA) x86-64` somente um
- `virtio-rng` (gerador virtual de números aleatórios)
- legacy AT teclado e PS/2 dispositivos de mouse
- dispositivo serial antigo

Modo de inicialização (UEFI and BIOS) suporte para AL2 023 no KVM

A `x86-64` imagem é testada com os dois modelos legados BIOS and UEFI modos de inicialização. As `aarch64` imagens são testadas com UEFI modo de inicialização.

> **Note**
>
> Por padrão, ao usar UEFI No modo de inicialização, alguns gerenciadores de máquinas virtuais provisionarão a VM com as chaves Microsoft Secure Boot, que habilitam a Inicialização Segura. Essa configuração não inicializará AL2 023.
> Como o carregador de inicialização AL2 023 não é assinado pela Microsoft, a VM deve ser provisionada sem chaves UEFI ou com as AL2 chaves 023 para inicialização segura.

> **Important**
>
> Suporte de inicialização segura para KVMas imagens ainda não foram validadas.

Limitações executando AL2 023 no KVM

Existem algumas limitações conhecidas na execução do AL2 023 no KVM.

> **Note**
>
> O código que implementa algumas das funcionalidades não suportadas listadas pode existir em AL2 023 e funcionar corretamente. A lista de funcionalidades não suportadas existe para que você possa tomar decisões informadas sobre em que confiar trabalhando hoje e o que a equipe do Amazon Linux qualificará como parte de futuras atualizações.

Limitações conhecidas com a execução de AL2 023 no KVM

- O agente convidado KVM não está atualmente empacotado ou não é compatível.
- Não há suporte para conexão e desconexão automática de CPU, memória ou qualquer outro tipo de dispositivo.
- A hibernação da VM não é suportada.
- A migração de VM não é suportada.
- A passagem de qualquer dispositivo, como PCI Passthrough ou USB Passthrough, não é compatível.

Requisitos para executar o AL2 023 em VMware

Esta seção descreve os requisitos para executar o AL2 023 em VMware. O VMware imagens de AL2 023 estão disponíveis somente para a `x86-64` arquitetura. VMware as imagens para não `aarch64` estão disponíveis ou não são suportadas. Esses requisitos são adicionais à base AL2023 requisitos do sistema para o VMware imagens.

Tópicos

- VMware requisitos de host para executar AL2 023 em VMware
- Suporte de dispositivo para AL2 023 em VMware
- Modo de inicialização (UEFI and BIOS) suporte para AL2 203 em VMware
- Limitações executando AL2 0.23 em VMware

VMware requisitos de host para executar AL2 023 em VMware

O AL2 023 VMware Atualmente, as imagens OVA são qualificadas no seguinte:

- VMware Estação de trabalho 17.5.0 em execução em hosts usando um processador Intel (R) Xeon (R) Platinum 8124M
- VMware vSphere 8.0 usando um processador Intel (R) Xeon (R) Platinum 8275CL

O AL2 023 VMware As imagens OVA especificam uma versão de hardware de máquina de 13.

VMware A versão 13 do hardware da máquina é suportada por:

- ESXi 6.5 ou posterior

- VMware Estação de trabalho 14 ou posterior

Suporte de dispositivo para AL2 023 em VMware

Os seguintes exemplos de VMware modelos de dispositivos foram testados para uso com AL2 023 VMware Imagens OVA (**`x86-64`**somente):

- `vmw_pvscsi` (VMware paravirtualizado SCSI controlador)
- `vmxnet3` (VMware dispositivo de rede paravirtualizado)
- `ata_piix`(legado IDE para uso somente com a unidade de CD-ROM virtual)

Adicional VMware modelos de dispositivos habilitados em AL2 023 VMware qualificação de imagem, mas não muito exercida:

- `vmw_vmcie` `vsock` interface relacionada (transporte de soquete virtual para o VMware agente convidado)
- Dispositivo de balão de memória `vmw_balloon`
- VMware `SVGA`controlador
- legacy AT teclado e PS/2 dispositivos de mouse

A ferramenta VMware o pacote guest agent (`open-vm-tools`) está disponível e instalado por padrão no AL2 023 VMware Imagens OVA.

Modo de inicialização (UEFI and BIOS) suporte para AL2 203 em VMware

A partir da versão 2023.3.20231211, o 023 AL2 VMware A imagem OVA foi validada em ambos os sistemas legados BIOS and UEFI modos de inicialização. A configuração padrão do OVA ainda é antiga BIOS mas pode ser alterado pelo usuário.

> **Important**
>
> O suporte ao Secure Boot requer UEFI, que não foi validado para AL2 023 em execução em VMware.

Limitações executando AL2 0.23 em VMware

Existem algumas limitações conhecidas na execução do AL2 023 em VMware.

> **Note**
>
> O código que implementa algumas das funcionalidades não suportadas listadas pode existir em AL2 023 e funcionar corretamente. A lista de funcionalidades não compatíveis existe para que os clientes possam tomar decisões informadas sobre em que confiar para trabalhar hoje e o que a equipe do Amazon Linux qualificará como parte de futuras atualizações.

Limitações conhecidas com a execução de AL2 023 em VMware

- UEFI O Secure Boot não está atualmente validado com AL2 023 ativado VMware.
- Não há suporte para conexão e desconexão automática de CPU, memória ou qualquer outro tipo de dispositivo.
- A hibernação da VM não é suportada.
- A migração de VM não é suportada.
- A passagem de qualquer dispositivo, como PCI Passthrough ou USB Passthrough, não é compatível.

Requisitos para executar o Amazon Linux 2023 no Hyper-V

Esta seção aborda os requisitos para executar o Amazon Linux 2023 no Hyper-V. As imagens do Hyper-V de AL2 023 estão disponíveis somente para a arquitetura. `x86-64` As imagens do Hyper-V para não `aarch64` estão disponíveis nem são suportadas no momento.

Esta seção aborda os requisitos adicionais, além da base, AL2023 requisitos do sistema para as imagens do Hyper-V.

Tópicos

- Requisitos de host Hyper-V para executar o Amazon Linux 2023 no Hyper-V
- Suporte de dispositivos para Amazon Linux 2023 no Hyper-V
- Limitações ao executar o Amazon Linux 2023 no Hyper-V

Requisitos de host Hyper-V para executar o Amazon Linux 2023 no Hyper-V

A principal qualificação do Amazon Linux 2023 no Hyper-V acontece no Windows Server 2022 executado em uma EC2 `c5.metal` instância.

Suporte de dispositivos para Amazon Linux 2023 no Hyper-V

O Amazon Linux 2023 é testado em máquinas virtuais Hyper-V de geração 1 e geração 2 com o seguinte conjunto de hardware virtualizado:

- VM de primeira geração (inicialização antiga do BIOS)
- VM de segunda geração (inicialização UEFI - Sem inicialização segura)
- Os seguintes modelos de dispositivos foram testados para uso com imagens AL2 023 do Hyper-V:
  - Armazenamento virtual Hyper-V *`hv_storvsc`* para o disco raiz e a unidade de CD-ROM emulada na geração 2 VMs
  - IDE PIIX emulado *`ata_piix`* para a unidade de CD-ROM virtual na Geração 1 VMs
  - Ethernet virtual Hyper-V `hv_netvsc`
- Os seguintes modelos de dispositivos estão habilitados, mas testados levemente:
  - Modo de texto VGA antigo na geração 1 VMs
  - Framebuffer *`simpledrmfb`* baseado em firmware UEFI na geração 2 VMs
  - Balão Hyper-V `hv_balloon`
  - Balão Hyper-V `hv_balloon`
  - Rato Hyper-V HID/Mouse `hid_hyperv`
- Os seguintes modos de dispositivo não estão habilitados no AL2 023 no momento:
  - Passagem PCI Hyper-V
  - Gráficos DRM Hyper-V

> **Important**
>
> Para máquinas virtuais de segunda geração, o Secure Boot não é suportado e deve ser desativado antes de iniciar a máquina virtual para uma inicialização bem-sucedida do Amazon Linux 2023. Atualmente, o Hyper-V suporta apenas o Secure Boot com componentes de software assinados pelas próprias chaves da Microsoft, enquanto o

bootloader do Amazon Linux é assinado por uma chave privada da Amazon. O Hyper-V não suporta a importação de chaves de terceiros no momento.

Limitações ao executar o Amazon Linux 2023 no Hyper-V

A seguir estão algumas limitações conhecidas na execução do Amazon Linux 2023 no Hyper-V:

> **Note**
>
> O código que implementa algumas das funcionalidades não suportadas listadas pode existir em AL2 023 e funcionar corretamente. A lista de funcionalidades não compatíveis existe para que os clientes possam tomar decisões informadas sobre em que confiar para trabalhar hoje e o que a equipe do Amazon Linux qualificará como parte de futuras atualizações.

Limitações conhecidas com a execução de AL2 023 no Hyper-V

- O modo UEFI Secure Boot não é atualmente suportado nem funciona com AL2 023 no Hyper-V
- Não há suporte para conexão e desconexão automática de CPU, memória ou qualquer outro tipo de dispositivo.
- A hibernação da máquina virtual (VM) não é compatível.
- A migração de máquina virtual (VM) não é compatível.
- A passagem de qualquer dispositivo, como PCI Passthrough ou USB Passthrough, não é compatível.

`cloud-init` Instalação e configuração do Amazon Linux 2023 quando usado fora da Amazon EC2

Esta seção aborda como instalar e configurar uma máquina virtual Amazon Linux 2023 quando não é executada diretamente na Amazon EC2, como quando em KVM ou VMware Hyper-V.

Por padrão, as imagens de uma máquina virtual Amazon Linux 2023 não vêm provisionadas com nenhuma senha de usuário ou chave ssh e obterão sua configuração de rede via DHCP na primeira interface de rede descoberta. Isso significa que, por padrão, sem configuração adicional, não há como se conectar à máquina virtual resultante.

Portanto, alguma forma de configuração precisa ser fornecida à máquina virtual. O mecanismo padrão para fazer isso no Amazon Linux é por meio de fontes de dados `cloud-init`.

O Amazon Linux 2023 foi qualificado com as seguintes fontes de dados:

NoCloud

Esse é o método tradicional de configuração de imagens locais por meio de um CD-ROM virtual contendo uma semente ISO9660 imagem com arquivos `cloud-init` de configuração.

VMware

Além disso, o Amazon Linux 2023 oferece suporte à configuração de VMware imagens em execução no vSphere por meio VMware da fonte VMware `guestinfo.userdata` de dados específica via e. `guestinfo.metadata`

> **Note**
>
> A configuração das fontes de dados pode ser diferente da do Amazon Linux 2. Mais especificamente, o Amazon Linux 2023 usa `systemd-networkd` para sua configuração e exige o uso do `cloud-init` “Networking Config Version 2", conforme documentado na documentação de configuração de `cloud-init` rede.

A documentação completa dos mecanismos de `cloud-init` configuração da versão `cloud-init` empacotada no Amazon Linux 2023 pode ser encontrada na documentação upstream `cloud-init`.

NoCloud (**`seed.iso`**) **`cloud-init`** configuração para Amazon Linux 2023 em KVM e VMware

Esta seção aborda como criar e usar uma `seed.iso` imagem para configurar o Amazon Linux 2023 em execução no KVM or VMware. Porque KVM and VMware ambientes não têm Amazon EC2 Instance Meta Data Service (IMDS), é necessário um método alternativo de configuração do Amazon Linux 2023, e fornecer uma `seed.iso` imagem é um desses métodos.

A imagem de `seed.iso` inclui as informações de configuração inicial necessárias para inicializar e configurar sua nova máquina virtual, como a configuração de rede, o nome do host e os dados do usuário.

> **Note**
>
> A imagem de `seed.iso` inclui somente as informações de configuração necessárias para inicializar a VM. Ela não inclui os arquivos do sistema operacional do Amazon Linux 2023.

Para gerar a imagem de `seed.iso`, você precisa de pelo menos dois arquivos de configuração:

`meta-data`

Esse arquivo normalmente inclui o nome do host da máquina virtual.

`user-data`

Esse arquivo normalmente configura contas de usuário, suas senhas, ssh pares de chaves e/ou mecanismos de acesso. Por padrão, o KVM e as VMware imagens do Amazon Linux 2023 criam uma conta de `ec2-user` usuário. Você pode usar o arquivo de configuração `user-data` para definir a senha e/ou as teclas SSH para esta conta de usuário padrão.

`network-config` (Opcional)

Esse arquivo normalmente fornece uma configuração de rede para a máquina virtual que substituirá a padrão. A configuração padrão é usar DHCP na primeira interface de rede disponível.

Crie a imagem de disco **`seed.iso`**

1. Em um computador Linux ou macOS, crie uma nova pasta chamada `seedconfig` e navegue até ela.

   > **Note**
   >
   > É possível usar o Windows ou outro sistema operacional para concluir essas etapas, mas você precisará encontrar a ferramenta equivalente a `mkisofs` para concluir a criação da imagem `seed.iso`.

2. Crie o arquivo de configuração `meta-data`.

   a. Crie um novo arquivo chamado `meta-data`.

b. Abra o `meta-data` arquivo usando seu editor preferido e adicione o seguinte, *vm-hostname* substituindo pelo nome do host da VM:

```
#cloud-config
local-hostname: vm-hostname
```

c. Salve e feche o arquivo de configuração `meta-data`.

3. Crie o arquivo de configuração `user-data`.

a. Crie um novo arquivo chamado `user-data`.

b. Abra o arquivo `user-data` usando o editor de texto de sua preferência e adicione o seguinte, fazendo as substituições necessárias:

```
#cloud-config
#vim:syntax=yaml
users:
# A user by the name 'ec2-user' is created in the image by default.
  - default
  - name: ec2-user
ssh_authorized_keys:
  - ssh-rsa ssh-key
# In the above line, replace ssh key with the content of your ssh public key.
```

c. Opcionalmente, você pode adicionar mais contas de usuário ao arquivo `user-data` de configuração.

Também é possível especificar contas de usuário adicionais, seus mecanismos de acesso, senhas e pares de chave. Para obter mais informações sobre as diretivas compatíveis, consulte a Documentação upstream de `cloud-init`.

d. Salve e feche o arquivo de configuração `user-data`.

4. (Opcional) Crie o arquivo de configuração `network-config`.

a. Crie um novo arquivo chamado `network-config`.

b. Abra o arquivo `network-config` usando o editor de texto de sua preferência e adicione o seguinte: substitua os vários endereços IP pelos apropriados para sua configuração.

```
#cloud-config
version: 2
```

```
ethernets:
  enp1s0:
    addresses:
      - 192.168.122.161/24
    gateway4: 192.168.122.1
    nameservers:
      addresses: 192.168.122.1
```

> **Note**
>
> `cloud-init`a configuração de rede fornece mecanismos para comparar com o MAC endereço da interface em vez de especificar o nome da interface, que pode mudar dependendo da configuração da VM. Esses (e mais) recursos `cloud-init` para configuração de rede são descritos com mais detalhes na documentação upstream do `cloud-init` Network Config Versão 2.

c. Salve e feche o arquivo de configuração `network-config`.

5. Crie a imagem do disco `seed.iso` usando os arquivos de configuração `meta-data`, `user-data` e `network-config` opcionais criados nas etapas anteriores.

   Siga um destes procedimentos, dependendo do sistema operacional no qual você está criando a imagem do disco `seed.iso`.

   - Em sistemas Linux, use uma ferramenta como **`mkisofs`** ou **`genisoimage`** para criar o arquivo completo `seed.iso`. Navegue até a pasta `seedconfig` e execute o comando a seguir:

     ```
     $ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data
     ```

   - Se você usar a `network-config`, inclua-a na invocação de **`mkisofs`**:

     ```
     $ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data
      network-config
     ```

   - Nos sistemas macOS, você pode usar uma ferramenta como **`hdiutil`** para gerar o arquivo `seed.iso` finalizado. Como **`hdiutil`** leva um nome de caminho em vez de uma lista de arquivos, a mesma invocação pode ser usada, independentemente de um arquivo de configuração `network-config` ter sido criado ou não.

```
$ hdiutil makehybrid -o seed.iso -hfs -joliet -iso -default-volume-name cidata
 seedconfig/
```

6. O `seed.iso` arquivo resultante agora pode ser anexado à sua nova máquina virtual Amazon Linux 2023 usando uma unidade de CD-ROM virtual `cloud-init` para localizar na primeira inicialização e aplicar a configuração ao sistema.

VMware**`cloud-init`**configuração guestinfo para AL2 023 em VMware

VMware ambientes não têm o Amazon EC2 Instance Meta Data Service (IMDS), portanto, é necessário um método alternativo de configuração do AL2 023. Esta seção descreve como usar um mecanismo de configuração alternativo para a unidade de CD-ROM `seed.iso` virtual que está disponível no VMware vSphere.

Esse método de configuração usa o VMware `extraconfig`mecanismo para fornecer dados de configuração para`cloud-init`. Para cada uma das chaves a seguir, uma ***`keyname`*`.encoding`** propriedade correspondente deve ser fornecida.

As seguintes chaves podem ser fornecidas para o VMware `extraconfig`mecanismo.

`guestinfo.metadata`

JSON or YAML contendo `cloud-init` metadados

`guestinfo.userdata`

A YAML documento contendo `cloud-init` dados do usuário no `cloud-config` formato.

`guestinfo.vendordata` (Opcional)

YAML contendo dados do `cloud-init` fornecedor

As propriedades de codificação correspondentes (`guestinfo.metadata.encoding`, `guestinfo.userdata.encoding` e `guestinfo.vendordata.encoding`) podem conter:

`base64`

O conteúdo da propriedade é codificado em `base64`.

`gzip+base64`

O conteúdo da propriedade é compactado com `gzip` após a codificação de `base64`.

> **Note**
>
> O `seed.iso` método oferece suporte a um arquivo de `network-config` configuração separado (opcional). VMware `guestinfo`difere na forma como a configuração de rede é fornecida. Informações adicionais são fornecidas na seção a seguir.

Se uma configuração de rede explícita for desejada, ela deverá ser incorporada `metadata` na forma de duas YAML or JSON propriedades:

`network`

Contém a configuração de rede codificada no formato JSON ou YAML.

`network.encoding`

Contém a codificação dos dados de configuração de rede acima. As codificações `cloud-init` compatíveis são as mesmas dos dados de `guestinfo`: `base64` e `gzip+base64`.

Example Usar o VMware Ferramenta vSphere **`govc`** CLI para passar a configuração com **`guestinfo`**

1. Prepare os arquivos de configuração `meta-data`, `user-data`, e os arquivos `network-config` de configuração opcionais conforme descrito emNoCloud (`seed.iso`) `cloud-init` configuração para Amazon Linux 2023 em KVM e VMware.
2. Converta os arquivos de configuração em formatos utilizáveis por VMware `guestinfo`.

```
# 'meta-data', `user-data` and `network-config` are the configuration
# files in the same format that would be used by a NoCloud (seed.iso)
# data source, read-them and convert them to VMware guestinfo
#
# The VM_NAME variable is assumed to be set to the name of the VM
# It is assumed that the necessary govc environment (credentials etc...) are
 already set

metadata=$(cat "meta-data")
userdata=$(cat "user-data")
if [ -e "network-config" ] ; then
    # We need to embed the network config inside the meta-data
    netconf=$(base64 -w0 "network-config")
```

```
    metadata=$(printf "%s\nnetwork: %s\nnetwork.encoding: base64" "$metadata"
 "$netconf")
fi
metadata=$(base64 -w0 <<< "$metadata")
govc vm.change -vm "$VM_NAME" \
    -e guestinfo.metadata="$metadata" \
    -e guestinfo.metadata.encoding="base64"
userdata=$(base64 -w0 <<< "$userdata")
govc vm.change -vm "$VM_NAME" \
    -e guestinfo.userdata="$userdata" \
    -e guestinfo.userdata.encoding="base64"
```

Comparando pacotes instalados na AMI padrão Amazon Linux 2023 com a imagem AL2 KVM 023

Uma comparação do RPMs presente na AMI padrão AL2 023 em comparação com o RPMs presente na imagem KVM AL2 023.

Pacote	AMI	KVM
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	
amazon-ec2-net-utils	2.5.1	
amazon-linux-onprem		1.2
amazon-linux-repo-cdn		2023.6.20241031
amazon-linux-repo-s3	2023.6.20241031	
amazon-linux-sb-keys	2023.1	2023.1

Pacote	AMI	KVM
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3.3.987.0	3.3.987.0
amd-ucode-firmware	20210208 (noarca)	20210208 (noarca)
at	3.1.23	3.1.23
attr	2.5.1	2.5.1
audit	3.0.6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2,0	
awscli-2	2.15.30	2.15.30
basesystem	11	11
bash	5.2.15	5.2.15
bash-completion	2.11	2.11
bc	1.07.1	1.07.1
bind-libs	9.18.28	9.18.28
bind-license	9.18.28	9.18.28
bind-utils	9.18.28	9.18.28
binutils	2,39	2,39
boost-filesystem	1.75.0	1.75.0
boost-system	1.75.0	1.75.0

Pacote	AMI	KVM
boost-thread	1.75.0	1.75.0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.68	2023.2.68
c-ares	1.19.1	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	22.2.2	22.2.2
cloud-init-cfg-ec2	22.2.2	
cloud-init-cfg-onprem		22.2.2
cloud-utils-growpart	0,31	0,31
coreutils	8,32	8,32
coreutils-common	8,32	8,32
cpio	2.13	2.13
cracklib	2.9.6	2.9.6
cracklib-dicts	2.9.6	2.9.6
crontabs	1.11	1.11
crypto-policies	2020428	2020428

Pacote	AMI	KVM
crypto-policies-scripts	2020428	2020428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27
cyrus-sasl-plain	2.1.27	2.1.27
dbus	1.12.28	1.12.28
dbus-broker	32	32
dbus-common	1.12.28	1.12.28
dbus-libs	1.12.28	1.12.28
device-mapper	1.02.185	1.02.185
device-mapper-libs	1.02.185	1.02.185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0
dnf-plugin-support-info	1.2	1.2

Pacote	AMI	KVM
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0,14	0,14
dyninst	10.2.1	10.2.1
e2fsprogs	1.46,5	1.46,5
e2fsprogs-libs	1.46,5	1.46,5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38
efivar-libs	38	38

Pacote	AMI	KVM
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5.15	5.15
expat	2.5.0	2.5.0
file	5,39	5,39
file-libs	5,39	5,39
filesystem	3,14	3,14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse-libs	2.9.9	2.9.9
gawk	5.1.0	5.1.0
gdbm-libs	1,19	1,19
gdisk	1.0.8	1.0.8
gettext	0,21	0,21
gettext-libs	0,21	0,21
ghc-srpm-macros	1.5.0	1.5.0

Pacote	AMI	KVM
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-all-langpacks	2.34	2.34
glibc-common	2.34	2.34
glibc-gconv-extra	2.34	2.34
glibc-locale-source	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1.20.7	1.20.7
grep	3.8	3.8
groff-base	1.22.4	1.22.4
grub2-common	2.06	2.06
grub2-efi-aa64-ec2	2.06 (64 de março)	2.06 (64 de março)
grub2-efi-x64-ec2	2,06 (x86_64)	2,06 (x86_64)
grub2-pc		2,06 (x86_64)
grub2-pc-modules	2.06	2.06 (março)
grub2-tools	2.06	2.06

Pacote	AMI	KVM
grub2-tools-minimal	2.06	2.06
grubby	8,40	8,40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3.23	3.23
hunspell	1.7.0	1.7.0
hunspell-en	0.20140811.1	0.20140811.1
hunspell-en-GB	0.20140811.1	0.20140811.1
hunspell-en-US	0.20140811.1	0.20140811.1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0,384	0,384
info	6.7	6.7
inih	49	49
initscripts	10.09	10.09
iproute	6.10.0	6.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0
jansson	2.14	2.14
jemalloc	5.2.1	5.2.1
jitterentropy	3.4.1	3.4.1

Pacote	AMI	KVM
jq	1.7.1	
json-c	0,14	0,14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6.1.112	6.1.112
kernel-libbpf	6.1.112	6.1.112
kernel-livepatch-repo-cdn		2023.6.20241031
kernel-livepatch-repo-s3	2023.6.20241031	
kernel-modules-extra		6.1.112
kernel-modules-extra-common		6.1.112
kernel-srpm-macros	1.0	1,0
kernel-tools	6.1.112	6.1.112
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29
kmod-libs	29	29
kpatch-runtime	0.9.7	0.9.7
krb5-libs	1.21.3	1.21.3

Pacote	AMI	KVM
less	608	608
libacl	2.3.1	2.3.1
libaio	0.3.111	0.3.111
libarchive	3.7.4	3.7.4
libargon2	27 de dezembro de 2017	27 de dezembro de 2017
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.37.4	2.37.4
libcap	2,48	2,48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.46,5	1.46,5
libcomps	0.1.20	0.1.20
libconfig	1.7.2	1.7.2
libcurl-minimal	8.5.0	8.5.0
libdb	5.3.28	5.3.28
libdhash	0.5.0	
libdnf	0.69,0	0.69,0

Pacote	AMI	KVM
libeconf	0.4.0	0.4.0
libedit	3.1	3.1
libev	4,33	4,33
libevent	2.1.12	2.1.12
libfdisk	2.37.4	2.37.4
libffi	3.4.4	3.4.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1,42	1,42
libibverbs	48,0	48,0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmaccalc	1.4.0	1.4.0
libldb	2.6.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.4	1.0.4

Pacote	AMI	KVM
libmodulemd	2.13.0	2.13.0
libmount	2.37.4	2.37.4
libnfsidmap	2.5.4	2.5.4
libnghttp2	1.59.0	1.59.0
libnl3	3.5.0	3.5.0
libpath_utils	0.2.1	0.2.1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0.21.1	0.21.1
libpwquality	1.4.4	1.4.4
libref_array	0.1.5	0.1.5
librepo	1.14.5	1.14.5
libreport-filesystem	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4
libselinux-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13

Pacote	AMI	KVM
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22
libss	1.46,5	1.46,5
libsss_certmap	2.9.4	
libsss_idmap	2.9.4	2.9.4
libsss_nss_idmap	2.9.4	2.9.4
libsss_sudo	2.9.4	
libstdc++	11.4.1	11.4.1
libstoragemgmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4.19.0	4.19.0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0,21	0,21
libtirpc	1.3.3	1.3.3
libunistring	0.9.10	0.9.10
libuser	0,63	0,63
libutempter	1.2.1	1.2.1
libuuid	2.37.4	2.37.4
libuv	1.47.0	1.47.0

Pacote	AMI	KVM
libverto	0.3.2	0.3.2
libverto-libev	0.3.2	0.3.2
libxcrypt	4.4.3	4.4.3
libxml2	2.10.4	2.10.4
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
linux-firmware-whence	20210208 (noarca)	20210208 (noarca)
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0.9.29	0.9.29
logrotate	3.20.1	3.20.1
lsof	4.94.0	4.94.0
lua-libs	5.4.4	5.4.4
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2.9.3	2.9.3
man-pages	5.10	5.10
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)
mpfr	4.1.0	4.1.0
nano	5,8	5,8
ncurses	6.2	6.2

Pacote	AMI	KVM
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2,0	2.0
newt	0,52,21	0,52,21
nfs-utils	2.5.4	2.5.4
npth	1,6	1.6
nspr	4.35.0	4.35.0
nss	3.90,0	3.90,0
nss-softokn	3.90,0	3.90,0
nss-softokn-freebl	3.90,0	3.90,0
nss-sysinit	3.90,0	3.90,0
nss-util	3.90,0	3.90,0
ntsysv	1.15	1.15
numactl-libs	2.0.14	2.0.14
ocaml-srpm-macros	6	6
oniguruma	6.9.7.1	
openblas-srpm-macros	2	2
openldap	2.4.57	2.4.57
openssh	8,7p1	8,7p1

Pacote	AMI	KVM
openssh-clients	8,7p1	8,7p1
openssh-server	8,7p1	8,7p1
openssl	3.0.8	3.0.8
openssl-libs	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12
os-prober	1,7	1,7
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0,80	0,80
pciutils	3.7.0	3.7.0
pciutils-libs	3.7.0	3.7.0
pcre2	10h40	10h40
pcre2-syntax	10h40	10h40
perl-Carp	1,50	1,50
perl-Class-Struct	0,66	0,66
perl-constant	1,33	1,33
perl-DynaLoader	1,47	1,47

Pacote	AMI	KVM
perl-Encode	3,15	3,15
perl-Errno	1,30	1,30
perl-Exporter	5,74	5,74
perl-Fcntl	1.13	1.13
perl-File-Basename	2,85	2,85
perl-File-Path	2,18	2,18
perl-File-stat	1,09	1,09
perl-File-Temp	0.231.100	0.231.100
perl-Getopt-Long	2,52	2,52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0,078	0,078
perl-if	0.60.800	0.60.800
perl-interpreter	5.32.1	5.32.1
perl-IO	1,43	1,43
perl-IPC-Open3	1,21	1,21
perl-libs	5.32.1	5.32.1
perl-MIME-Base64	3.16	3.16
perl-mro	1,23	1,23
perl-overload	1,31	1,31
perl-overloading	0,02	0,02

Pacote	AMI	KVM
perl-parent	0,238	0,238
perl-PathTools	3,78	3,78
perl-Pod-Escapes	1,07	1,07
perl-podlators	4.14	4.14
perl-Pod-Perldoc	3.28.01	3.28.01
perl-Pod-Simple	3,42	3,42
perl-Pod-Usage	2.01	2.01
perl-POSIX	1,94	1,94
perl-Scalar-List-Utils	1,56	1,56
perl-SelectSaver	1.02	1.02
perl-Socket	2.032	2.032
perl-srpm-macros	1	1
perl-Storable	3.21	3.21
perl-subs	1,03	1,03
perl-Symbol	1,08	1,08
perl-Term-ANSIColor	5.01	5.01
perl-Term-Cap	1.17	1.17
perl-Text-ParseWords	3,30	3,30
perl-Text-Tabs+Wrap	2021.07.26	2021.07.26
perl-Time-Local	1.300	1.300

Pacote	AMI	KVM
perl-vars	1,05	1,05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1,18	1,18
procps-ng	3.3.17	3.3.17
protobuf-c	1.4.1	1.4.1
psacct	6.6.4	6.6.4
psmisc	23,4	23,4
publicsuffix-list-dafsa	2024/02/12	2024/02/12
python3	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0
python3-audit	3.0.6	3.0.6
python3-awscrt	0.19.19	0.19.19
python3-babel	2.9.1	2.9.1
python3-cffi	1.14.5	1.14.5
python3-chardet	4.0.0	4.0.0

Pacote	AMI	KVM
python3-colorama	0.4.4	0.4.4
python3-configobj	5.0.6	5.0.6
python3-cryptography	36.0.1	36.0.1
python3-daemon	2.3.0	
python3-dateutil	2.8.1	2.8.1
python3-dbus	1.2.18	1.2.18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0,16	0,16
python3-gpg	1.15.1	1.15.1
python3-hawkey	0.69,0	0.69,0
python3-idna	(2.10)	(2.10)
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1,21	1,21
python3-jsonpointer	2,0	2.0
python3-jsonschema	3.2.0	3.2.0
python3-libcomps	0.1.20	0.1.20
python3-libdnf	0.69,0	0.69,0

Pacote	AMI	KVM
python3-libs	3.9.16	3.9.16
python3-libselinux	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0.10.6	0.10.6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21.3.1	21.3.1
python3-ply	3.11	3.11
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-too lkit	3.0.24	3.0.24
python3-pycparser	2.20	2.20
python3-pyrsistent	0.17.3	0.17.3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022.7.1	2022.7.1

Pacote	AMI	KVM
python3-pyyaml	5.4.1	5.4.1
python3-requests	2.25.1	2.25.1
python3-rpm	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6
python3-ruamel-yaml-clib	0.1.2	0.1.2
python3-setools	4.4.1	4.4.1
python3-setuptools	59.6.0	59.6.0
python3-setuptools-wheel	59.6.0	59.6.0
python3-six	1.15.0	1.15.0
python3-systemd	235	235
python3-urllib3	1.25.10	1.25.10
python3-wcwidth	0.2.5	0.2.5
python-chevron	0.13.1	
python-srpm-macros	3.9	3.9
quota	4.06	4.06
quota-nls	4.06	4.06
readline	8.1	8.1
rng-tools	6.14	6.14
rootfiles	8.1	8.1

Pacote	AMI	KVM
rpcbind	1.2.6	1.2.6
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	4.16.1.3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0
sed	4.8	4.8
selinux-policy	38.1.45	38.1.45
selinux-policy-targeted	38.1.45	38.1.45
setup	2.13.7	2.13.7
shadow-utils	4,9	4,9
slang	2.3.2	2.3.2
sqlite-libs	3.40,0	3.40,0
sssd-client	2.9.4	2.9.4

Pacote	AMI	KVM
sssd-common	2.9.4	
sssd-kcm	2.9.4	
sssd-nfs-idmap	2.9.4	
strace	6.8	6.8
sudo	1.9.15	1.9.15
sysctl-defaults	1.0	1,0
sysstat	12.5.6	12.5.6
systemd	252,23	252,23
systemd-libs	252,23	252,23
systemd-networkd	252,23	252,23
systemd-pam	252,23	252,23
systemd-resolved	252,23	252,23
systemd-udev	252,23	252,23
system-release	2023.6.20241031	2023.6.20241031
systemtap-runtime	4.8	4.8
tar	1,34	1,34
tbb	2020.3	2020.3
tcpdump	4.99.1	4.99.1
tcsh	24.6.07	24.6.07
time	1.9	1.9

Pacote	AMI	KVM
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2.37.4	2.37.4
util-linux-core	2.37.4	2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
wget	1.21.3	1.21.3
which	2.21	2.21
words	3.0	3.0
xfsdump	3.1.11	3.1.11
xfsprogs	5.18.0	5.18.0
xxd	9.0.2153	9.0.2153
xxhash-libs	0.8.0	0.8.0
xz	5.2.5	5.2.5

Pacote	AMI	KVM
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1.2.11	1.2.11
zram-generator	1.1.2	
zram-generator-defaults	1.1.2	
zstd	1.5.5	1.5.5

Comparando pacotes instalados na AMI padrão Amazon Linux 2023 com a imagem AL2 023 VMware OVA

Uma comparação do RPMs presente na AMI padrão AL2 023 em comparação com o RPMs presente na imagem AL2 023 VMware OVA.

Pacote	AMI	VMware ÓVULOS
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	
amazon-ec2-net-utils	2.5.1	
amazon-linux-onprem		1.2
amazon-linux-repo-cdn		2023.6.20241031

Pacote	AMI	VMware ÓVULOS
amazon-linux-repo-s3	2023.6.20241031	
amazon-linux-sb-keys	2023.1	2023.1
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3.3.987.0	3.3.987.0
amd-ucode-firmware	20210208	20210208
at	3.1.23	3.1.23
attr	2.5.1	2.5.1
audit	3.0.6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2,0	
awscli-2	2.15.30	2.15.30
basesystem	11	11
bash	5.2.15	5.2.15
bash-completion	2.11	2.11
bc	1.07.1	1.07.1
bind-libs	9.18.28	9.18.28
bind-license	9.18.28	9.18.28
bind-utils	9.18.28	9.18.28
binutils	2,39	2,39

Pacote	AMI	VMware ÓVULOS
boost-filesystem	1.75.0	1.75.0
boost-system	1.75.0	1.75.0
boost-thread	1.75.0	1.75.0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.68	2023.2.68
c-ares	1.19.1	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	22.2.2	22.2.2
cloud-init-cfg-ec2	22.2.2	
cloud-init-cfg-onprem		22.2.2
cloud-utils-growpart	0,31	0,31
coreutils	8,32	8,32
coreutils-common	8,32	8,32
cpio	2.13	2.13
cracklib	2.9.6	2.9.6
cracklib-dicts	2.9.6	2.9.6
crontabs	1.11	1.11

Pacote	AMI	VMware ÓVULOS
crypto-policies	2020428	2020428
crypto-policies-scripts	2020428	2020428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27
cyrus-sasl-plain	2.1.27	2.1.27
dbus	1.12.28	1.12.28
dbus-broker	32	32
dbus-common	1.12.28	1.12.28
dbus-libs	1.12.28	1.12.28
device-mapper	1.02.185	1.02.185
device-mapper-libs	1.02.185	1.02.185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0

Pacote	AMI	VMware ÓVULOS
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0,14	0,14
dyninst	10.2.1	10.2.1
e2fsprogs	1.46,5	1.46,5
e2fsprogs-libs	1.46,5	1.46,5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38

Pacote	AMI	VMware ÓVULOS
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5.15	5.15
expat	2.5.0	2.5.0
file	5,39	5,39
file-libs	5,39	5,39
filesystem	3.14	3.14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse3		3.10.4
fuse3-libs		3.10.4
fuse-common		3.10.4
fuse-libs	2.9.9	2.9.9
gawk	5.1.0	5.1.0
gdbm-libs	1,19	1,19

Pacote	AMI	VMware ÓVULOS
gdisk	1.0.8	1.0.8
gettext	0,21	0,21
gettext-libs	0,21	0,21
ghc-srpm-macros	1.5.0	1.5.0
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-all-langpacks	2.34	2.34
glibc-common	2.34	2.34
glibc-gconv-extra	2.34	2.34
glibc-locale-source	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1.20.7	1.20.7
grep	3.8	3.8
groff-base	1.22.4	1.22.4
grub2-common	2.06	2.06
grub2-efi-x64-ec2	2.06	2.06

Pacote	AMI	VMware ÓVULOS
grub2-pc		2.06
grub2-pc-modules	2.06	2.06
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8,40	8,40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3.23	3.23
hunspell	1.7.0	1.7.0
hunspell-en	0.20140811.1	0.20140811.1
hunspell-en-GB	0.20140811.1	0.20140811.1
hunspell-en-US	0.20140811.1	0.20140811.1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0,384	0,384
info	6.7	6.7
inih	49	49
initscripts	10.09	10.09
iproute	6.10.0	6.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0

Pacote	AMI	VMware ÓVULOS
jansson	2.14	2.14
jemalloc	5.2.1	5.2.1
jitterentropy	3.4.1	3.4.1
jq	1.7.1	
json-c	0,14	0,14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6.1.112	6.1.112
kernel-libbpf	6.1.112	6.1.112
kernel-livepatch-repo-cdn		2023.6.20241031
kernel-livepatch-repo-s3	2023.6.20241031	
kernel-modules-extra		6.1.112
kernel-modules-extra-common		6.1.112
kernel-srpm-macros	1.0	1,0
kernel-tools	6.1.112	6.1.112
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29

Pacote	AMI	VMware ÓVULOS
kmod-libs	29	29
kpatch-runtime	0.9.7	0.9.7
krb5-libs	1.21.3	1.21.3
less	608	608
libacl	2.3.1	2.3.1
libaio	0.3.111	0.3.111
libarchive	3.7.4	3.7.4
libargon2	27 de dezembro de 2017	27 de dezembro de 2017
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.37.4	2.37.4
libcap	2,48	2,48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.46,5	1.46,5
libcomps	0.1.20	0.1.20
libconfig	1.7.2	1.7.2
libcurl-minimal	8.5.0	8.5.0

Pacote	AMI	VMware ÓVULOS
libdb	5.3.28	5.3.28
libdhash	0.5.0	
libdnf	0.69,0	0.69,0
libeconf	0.4.0	0.4.0
libedit	3.1	3.1
libev	4,33	4,33
libevent	2.1.12	2.1.12
libfdisk	2.37.4	2.37.4
libffi	3.4.4	3.4.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1,42	1,42
libibverbs	48,0	48,0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmaccalc	1.4.0	1.4.0
libldb	2.6.2	

Pacote	AMI	VMware ÓVULOS
libmaxminddb	1.5.2	1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2.37.4	2.37.4
libmspack		0.10.1
libnfsidmap	2.5.4	2.5.4
libnghttp2	1.59.0	1.59.0
libnl3	3.5.0	3.5.0
libpath_utils	0.2.1	0.2.1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0,21,1	0,21,1
libpwquality	1.4.4	1.4.4
libref_array	0.1.5	0.1.5
librepo	1.14.5	1.14.5
libreport-filesystem	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4

Pacote	AMI	VMware ÓVULOS
libselinux-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22
libss	1.46,5	1.46,5
libsss_certmap	2.9.4	
libsss_idmap	2.9.4	2.9.4
libsss_nss_idmap	2.9.4	2.9.4
libsss_sudo	2.9.4	
libstdc++	11.4.1	11.4.1
libstoragemgmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4.19.0	4.19.0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0,21	0,21
libtirpc	1.3.3	1.3.3
libtool-ltdl		2.4.7

Pacote	AMI	VMware ÓVULOS
libunistring	0.9.10	0.9.10
libuser	0,63	0,63
libutempter	1.2.1	1.2.1
libuuid	2.37.4	2.37.4
libuv	1.47.0	1.47.0
libverto	0.3.2	0.3.2
libverto-libev	0.3.2	0.3.2
libxcrypt	4.4.3	4.4.3
libxml2	2.10.4	2.10.4
libxslt		1.1.34
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
linux-firmware-whence	20210208	20210208
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0.9.29	0.9.29
logrotate	3.20.1	3.20.1
lsof	4.94.0	4.94.0
lua-libs	5.4.4	5.4.4
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4

Pacote	AMI	VMware ÓVULOS
man-db	2.9.3	2.9.3
man-pages	5.10	5.10
microcode_ctl	2.1	2.1
mpfr	4.1.0	4.1.0
nano	5,8	5,8
ncurses	6.2	6.2
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2,0	2.0
newt	0,52,21	0,52,21
nfs-utils	2.5.4	2.5.4
npth	1,6	1.6
nspr	4.35.0	4.35.0
nss	3.90,0	3.90,0
nss-softokn	3.90,0	3.90,0
nss-softokn-freebl	3.90,0	3.90,0
nss-sysinit	3.90,0	3.90,0
nss-util	3.90,0	3.90,0
ntsysv	1.15	1.15

Pacote	AMI	VMware ÓVULOS
numactl-libs	2.0.14	2.0.14
ocaml-srpm-macros	6	6
oniguruma	6.9.7.1	
openblas-srpm-macros	2	2
openldap	2.4.57	2.4.57
openssh	8,7p1	8,7p1
openssh-clients	8,7p1	8,7p1
openssh-server	8,7p1	8,7p1
openssl	3.0.8	3.0.8
openssl-libs	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12
open-vm-tools		12.3.0
os-prober	1,7	1,7
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0,80	0,80
pciutils	3.7.0	3.7.0

Pacote	AMI	VMware ÓVULOS
pciutils-libs	3.7.0	3.7.0
pcre2	10h40	10h40
pcre2-syntax	10h40	10h40
perl-Carp	1,50	1,50
perl-Class-Struct	0,66	0,66
perl-constant	1,33	1,33
perl-DynaLoader	1,47	1,47
perl-Encode	3,15	3,15
perl-Errno	1,30	1,30
perl-Exporter	5,74	5,74
perl-Fcntl	1.13	1.13
perl-File-Basename	2,85	2,85
perl-File-Path	2,18	2,18
perl-File-stat	1,09	1,09
perl-File-Temp	0.231.100	0.231.100
perl-Getopt-Long	2,52	2,52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0,078	0,078
perl-if	0.60.800	0.60.800
perl-interpreter	5.32.1	5.32.1

Pacote	AMI	VMware ÓVULOS
perl-IO	1,43	1,43
perl-IPC-Open3	1,21	1,21
perl-libs	5.32.1	5.32.1
perl-MIME-Base64	3.16	3.16
perl-mro	1,23	1,23
perl-overload	1,31	1,31
perl-overloading	0,02	0,02
perl-parent	0,238	0,238
perl-PathTools	3,78	3,78
perl-Pod-Escapes	1,07	1,07
perl-podlators	4.14	4.14
perl-Pod-Perldoc	3.28.01	3.28.01
perl-Pod-Simple	3,42	3,42
perl-Pod-Usage	2.01	2.01
perl-POSIX	1,94	1,94
perl-Scalar-List-Utils	1,56	1,56
perl-SelectSaver	1.02	1.02
perl-Socket	2.032	2.032
perl-srpm-macros	1	1
perl-Storable	3.21	3.21

Pacote	AMI	VMware ÓVULOS
perl-subs	1,03	1,03
perl-Symbol	1,08	1,08
perl-Term-ANSIColor	5.01	5.01
perl-Term-Cap	1.17	1.17
perl-Text-ParseWords	3,30	3,30
perl-Text-Tabs+Wrap	2021.07.26	2021.07.26
perl-Time-Local	1.300	1.300
perl-vars	1,05	1,05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1,18	1,18
procps-ng	3.3.17	3.3.17
protobuf-c	1.4.1	1.4.1
psacct	6.6.4	6.6.4
psmisc	23,4	23,4
publicsuffix-list-dafsa	2024/02/12	2024/02/12

Pacote	AMI	VMware ÓVULOS
python3	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0
python3-audit	3.0.6	3.0.6
python3-awscrt	0.19.19	0.19.19
python3-babel	2.9.1	2.9.1
python3-cffi	1.14.5	1.14.5
python3-chardet	4.0.0	4.0.0
python3-colorama	0.4.4	0.4.4
python3-configobj	5.0.6	5.0.6
python3-cryptography	36.0.1	36.0.1
python3-daemon	2.3.0	
python3-dateutil	2.8.1	2.8.1
python3-dbus	1.2.18	1.2.18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0,16	0,16
python3-gpg	1.15.1	1.15.1
python3-hawkey	0.69,0	0.69,0
python3-idna	(2.10)	(2.10)

Pacote	AMI	VMware ÓVULOS
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1,21	1,21
python3-jsonpointer	2,0	2.0
python3-jsonschema	3.2.0	3.2.0
python3-libcomps	0.1.20	0.1.20
python3-libdnf	0.69,0	0.69,0
python3-libs	3.9.16	3.9.16
python3-libselinux	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0.10.6	0.10.6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21.3.1	21.3.1
python3-ply	3.11	3.11
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2

Pacote	AMI	VMware ÓVULOS
python3-prompt-toolkit	3.0.24	3.0.24
python3-pycparser	2.20	2.20
python3-pyrsistent	0.17.3	0.17.3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022.7.1	2022.7.1
python3-pyyaml	5.4.1	5.4.1
python3-requests	2.25.1	2.25.1
python3-rpm	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6
python3-ruamel-yaml-clib	0.1.2	0.1.2
python3-setools	4.4.1	4.4.1
python3-setuptools	59.6.0	59.6.0
python3-setuptools-wheel	59.6.0	59.6.0
python3-six	1.15.0	1.15.0
python3-systemd	235	235
python3-urllib3	1.25.10	1.25.10
python3-wcwidth	0.2.5	0.2.5

Pacote	AMI	VMware ÓVULOS
python-chevron	0.13.1	
python-srpm-macros	3.9	3.9
quota	4.06	4.06
quota-nls	4.06	4.06
readline	8.1	8.1
rng-tools	6.14	6.14
rootfiles	8.1	8.1
rpcbind	1.2.6	1.2.6
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	4.16.1.3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0
sed	4.8	4.8
selinux-policy	38.1.45	38.1.45

Pacote	AMI	VMware ÓVULOS
selinux-policy-targeted	38.1.45	38.1.45
setup	2.13.7	2.13.7
shadow-utils	4,9	4,9
slang	2.3.2	2.3.2
sqlite-libs	3.40,0	3.40,0
sssd-client	2.9.4	2.9.4
sssd-common	2.9.4	
sssd-kcm	2.9.4	
sssd-nfs-idmap	2.9.4	
strace	6.8	6.8
sudo	1.9.15	1.9.15
sysctl-defaults	1.0	1,0
sysstat	12.5.6	12.5.6
systemd	252,23	252,23
systemd-libs	252,23	252,23
systemd-networkd	252,23	252,23
systemd-pam	252,23	252,23
systemd-resolved	252,23	252,23
systemd-udev	252,23	252,23
system-release	2023.6.20241031	2023.6.20241031

Pacote	AMI	VMware ÓVULOS
systemtap-runtime	4.8	4.8
tar	1,34	1,34
tbb	2020.3	2020.3
tcpdump	4.99.1	4.99.1
tcsh	24.6.07	24.6.07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2.37.4	2.37.4
util-linux-core	2.37.4	2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153
wget	1.21.3	1.21.3
which	2.21	2.21

Pacote	AMI	VMware ÓVULOS
words	3.0	3.0
xfsdump	3.1.11	3.1.11
xfsprogs	5.18.0	5.18.0
xmlsec1		1.2.3
xmlsec1-openssl		1.2.3
xxd	9.0.2153	9.0.2153
xxhash-libs	0.8.0	0.8.0
xz	5.2.5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1.2.11	1.2.11
zram-generator	1.1.2	
zram-generator-defaults	1.1.2	
zstd	1.5.5	1.5.5

Comparando pacotes instalados na AMI padrão Amazon Linux 2023 com a imagem AL2 023 Hyper-V

Uma comparação do RPMs presente na AMI padrão AL2 023 em comparação com o RPMs presente na imagem AL2 023 Hyper-V.

Pacote	AMI	Hyper-V VHDX
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	
amazon-ec2-net-utils	2.4.1	
amazon-linux-onprem		1.2
amazon-linux-repo-cdn		2023.4.20240319
amazon-linux-repo-s3	2023.4.20240319	
amazon-linux-sb-keys	2023.1	2023.1
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3.2.2303.0	3.2.2303.0
at	3.1.23	3.1.23
attr	2.5.1	2.5.1
audit	3.0.6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2,0	
awscli-2	2.14.5	2.14.5
basesystem	11	11

Pacote	AMI	Hyper-V VHDX
bash	5.2.15	5.2.15
bash-completion	2.11	2.11
bc	1.07.1	1.07.1
bind-libs	9.16.48	9.16.48
bind-license	9.16.48	9.16.48
bind-utils	9.16.48	9.16.48
binutils	2,39	2,39
boost-filesystem	1.75.0	1.75.0
boost-system	1.75.0	1.75.0
boost-thread	1.75.0	1.75.0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023.2.64	2023.2.64
c-ares	1.19.0	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	22.2.2	22.2.2
cloud-init-cfg-ec2	22.2.2	
cloud-init-cfg-onprem		22.2.2

Pacote	AMI	Hyper-V VHDX
cloud-utils-growpart	0,31	0,31
coreutils	8,32	8,32
coreutils-common	8,32	8,32
cpio	2.13	2.13
cracklib	2.9.6	2.9.6
cracklib-dicts	2.9.6	2.9.6
crontabs	1.11	1.11
crypto-policies	2020428	2020428
crypto-policies-scripts	2020428	2020428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8.5.0	8.5.0
cyrus-sasl-lib	2.1.27	2.1.27
cyrus-sasl-plain	2.1.27	2.1.27
dbus	1.12.28	1.12.28
dbus-broker	32	32
dbus-common	1.12.28	1.12.28
dbus-libs	1.12.28	1.12.28
device-mapper	1.02.185	1.02.185
device-mapper-libs	1.02.185	1.02.185

Pacote	AMI	Hyper-V VHDX
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0,14	0,14
dyninst	10.2.1	10.2.1
e2fsprogs	1.46,5	1.46,5
e2fsprogs-libs	1.46,5	1.46,5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	

Pacote	AMI	Hyper-V VHDX
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5.15	5.15
expat	2.5.0	2.5.0
file	5,39	5,39
file-libs	5,39	5,39
filesystem	3.14	3.14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse-libs	2.9.9	2.9.9

Pacote	AMI	Hyper-V VHDX
gawk	5.1.0	5.1.0
gdbm-libs	1,19	1,19
gdisk	1.0.8	1.0.8
gettext	0,21	0,21
gettext-libs	0,21	0,21
ghc-srpm-macros	1.5.0	1.5.0
glib2	2.74.7	2.74.7
glibc	2.34	2.34
glibc-all-langpacks	2.34	2.34
glibc-common	2.34	2.34
glibc-gconv-extra	2.34	2.34
glibc-locale-source	2.34	2.34
gmp	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1.20.7	1.20.7
grep	3.8	3.8
groff-base	1.22.4	1.22.4

Pacote	AMI	Hyper-V VHDX
grub2-common	2.06	2.06
grub2-efi-x64-ec2	2.06	2.06
grub2-pc		2.06
grub2-pc-modules	2.06	2.06
grub2-tools	2.06	2.06
grub2-tools-minimal	2.06	2.06
grubby	8,40	8,40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3.23	3.23
hunspell	1.7.0	1.7.0
hunspell-en	0.20140811.1	0.20140811.1
hunspell-en-GB	0.20140811.1	0.20140811.1
hunspell-en-US	0.20140811.1	0.20140811.1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0,353	0,353
hyperv-daemons		0
hyperv-daemons-license		0
hypervfcopyd		0
hypervkvpd		0

Pacote	AMI	Hyper-V VHDX
hyperv-tools		0
hypervvssd		0
info	6.7	6.7
inih	49	49
initscripts	10.09	10.09
iproute	5.10.0	5.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0
jansson	2.14	2.14
jitterentropy	3.4.1	3.4.1
jq	1.7.1	1.7.1
json-c	0,14	0,14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6.1.79	6.1.79
kernel-livepatch-repo-cdn		2023.4.20240319
kernel-livepatch-repo-s3	2023.4.20240319	
kernel-modules-extra		6.1.79

Pacote	AMI	Hyper-V VHDX
kernel-modules-extra-common		6.1.79
kernel-srpm-macros	1.0	1,0
kernel-tools	6.1.79	6.1.79
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29
kmod-libs	29	29
kpatch-runtime	0.9.7	0.9.7
krb5-libs	1,21	1,21
less	608	608
libacl	2.3.1	2.3.1
libaio	0.3.111	0.3.111
libarchive	3.5.3	3.5.3
libargon2	27 de dezembro de 2017	27 de dezembro de 2017
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0.1.1	0.1.1
libblkid	2.37.4	2.37.4
libcap	2,48	2,48
libcap-ng	0.8.2	0.8.2

Pacote	AMI	Hyper-V VHDX
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1.46,5	1.46,5
libcomps	0.1.20	0.1.20
libconfig	1.7.2	1.7.2
libcurl-minimal	8.5.0	8.5.0
libdb	5.3.28	5.3.28
libdhash	0.5.0	
libdnf	0.69,0	0.69,0
libeconf	0.4.0	0.4.0
libedit	3.1	3.1
libev	4,33	4,33
libevent	2.1.12	2.1.12
libfdisk	2.37.4	2.37.4
libffi	3.4.4	3.4.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1,42	1,42

Pacote	AMI	Hyper-V VHDX
libibverbs	48,0	48,0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmaccalc	1.4.0	1.4.0
libldb	2.6.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0.1.3	0.1.3
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2.37.4	2.37.4
libnfsidmap	2.5.4	2.5.4
libnghttp2	1.57.0	1.57.0
libnl3	3.5.0	3.5.0
libpath_utils	0.2.1	0.2.1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0,21,1	0,21,1
libpwquality	1.4.4	1.4.4

Pacote	AMI	Hyper-V VHDX
libref_array	0.1.5	0.1.5
librepo	1.14.5	1.14.5
libreport-filesystem	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4
libselinux-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2.13	2.13
libsmartcols	2.37.4	2.37.4
libsolv	0.7.22	0.7.22
libss	1.46,5	1.46,5
libsss_certmap	2.9.4	
libsss_idmap	2.9.4	2.9.4
libsss_nss_idmap	2.9.4	2.9.4
libsss_sudo	2.9.4	
libstdc++	11.4.1	11.4.1
libstoragemgmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4.19.0	4.19.0

Pacote	AMI	Hyper-V VHDX
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0,21	0,21
libtirpc	1.3.3	1.3.3
libunistring	0.9.10	0.9.10
libuser	0,63	0,63
libutempter	1.2.1	1.2.1
libuuid	2.37.4	2.37.4
libuv	1.47.0	1.47.0
libverto	0.3.2	0.3.2
libverto-libev	0.3.2	0.3.2
libxcrypt	4.4.3	4.4.3
libxml2	2.10.4	2.10.4
libyaml	0.2.5	0.2.5
libzstd	1.5.5	1.5.5
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0.9.29	0.9.29
logrotate	3.20.1	3.20.1
lsof	4.94.0	4.94.0
lua-libs	5.4.4	5.4.4

Pacote	AMI	Hyper-V VHDX
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2.9.3	2.9.3
man-pages	5.10	5.10
microcode_ctl	2.1	2.1
mpfr	4.1.0	4.1.0
nano	5,8	5,8
ncurses	6.2	6.2
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2,0	2.0
newt	0,52,21	0,52,21
nfs-utils	2.5.4	2.5.4
npth	1,6	1.6
nspr	4.35.0	4.35.0
nss	3.90,0	3.90,0
nss-softokn	3.90,0	3.90,0
nss-softokn-freebl	3.90,0	3.90,0
nss-sysinit	3.90,0	3.90,0

Pacote	AMI	Hyper-V VHDX
nss-util	3.90,0	3.90,0
ntsysv	1.15	1.15
numactl-libs	2.0.14	2.0.14
ocaml-srpm-macros	6	6
oniguruma	6.9.7.1	6.9.7.1
openblas-srpm-macros	2	2
openldap	2.4.57	2.4.57
openssh	8,7p1	8,7p1
openssh-clients	8,7p1	8,7p1
openssh-server	8,7p1	8,7p1
openssl	3.0.8	3.0.8
openssl-libs	3.0.8	3.0.8
openssl-pkcs11	0.4.12	0.4.12
os-prober	1,7	1,7
p11-kit	0.24.1	0.24.1
p11-kit-trust	0.24.1	0.24.1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0,80	0,80

Pacote	AMI	Hyper-V VHDX
pciutils	3.7.0	3.7.0
pciutils-libs	3.7.0	3.7.0
pcre2	10h40	10h40
pcre2-syntax	10h40	10h40
perl-Carp	1,50	1,50
perl-Class-Struct	0,66	0,66
perl-constant	1,33	1,33
perl-DynaLoader	1,47	1,47
perl-Encode	3,15	3,15
perl-Errno	1,30	1,30
perl-Exporter	5,74	5,74
perl-Fcntl	1.13	1.13
perl-File-Basename	2,85	2,85
perl-File-Path	2,18	2,18
perl-File-stat	1,09	1,09
perl-File-Temp	0.231.100	0.231.100
perl-Getopt-Long	2,52	2,52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0,078	0,078
perl-if	0.60.800	0.60.800

Pacote	AMI	Hyper-V VHDX
perl-interpreter	5.32.1	5.32.1
perl-IO	1,43	1,43
perl-IPC-Open3	1,21	1,21
perl-libs	5.32.1	5.32.1
perl-MIME-Base64	3.16	3.16
perl-mro	1,23	1,23
perl-overload	1,31	1,31
perl-overloading	0,02	0,02
perl-parent	0,238	0,238
perl-PathTools	3,78	3,78
perl-Pod-Escapes	1,07	1,07
perl-podlators	4.14	4.14
perl-Pod-Perldoc	3.28.01	3.28.01
perl-Pod-Simple	3,42	3,42
perl-Pod-Usage	2.01	2.01
perl-POSIX	1,94	1,94
perl-Scalar-List-Utils	1,56	1,56
perl-SelectSaver	1.02	1.02
perl-Socket	2.032	2.032
perl-srpm-macros	1	1

Pacote	AMI	Hyper-V VHDX
perl-Storable	3.21	3.21
perl-subs	1,03	1,03
perl-Symbol	1,08	1,08
perl-Term-ANSIColor	5.01	5.01
perl-Term-Cap	1.17	1.17
perl-Text-ParseWords	3,30	3,30
perl-Text-Tabs+Wrap	2021.07.26	2021.07.26
perl-Time-Local	1.300	1.300
perl-vars	1,05	1,05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1,18	1,18
procps-ng	3.3.17	3.3.17
protobuf-c	1.4.1	1.4.1
psacct	6.6.4	6.6.4
psmisc	23,4	23,4

Pacote	AMI	Hyper-V VHDX
publicsuffix-list-dafsa	20240212	20240212
python3	3.9.16	3.9.16
python3-attrs	20.3.0	20.3.0
python3-audit	3.0.6	3.0.6
python3-awscrt	0.19.19	0.19.19
python3-babel	2.9.1	2.9.1
python3-cffi	1.14.5	1.14.5
python3-chardet	4.0.0	4.0.0
python3-colorama	0.4.4	0.4.4
python3-configobj	5.0.6	5.0.6
python3-cryptography	36.0.1	36.0.1
python3-daemon	2.3.0	
python3-dateutil	2.8.1	2.8.1
python3-dbus	1.2.18	1.2.18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0,16	0,16
python3-gpg	1.15.1	1.15.1

Pacote	AMI	Hyper-V VHDX
python3-hawkey	0.69,0	0.69,0
python3-idna	(2.10)	(2.10)
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1,21	1,21
python3-jsonpointer	2,0	2.0
python3-jsonschema	3.2.0	3.2.0
python3-libcomps	0.1.20	0.1.20
python3-libdnf	0.69,0	0.69,0
python3-libs	3.9.16	3.9.16
python3-libselinux	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0.10.6	0.10.6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21.3.1	21.3.1
python3-ply	3.11	3.11

Pacote	AMI	Hyper-V VHDX
python3-policycoreutils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-toolkit	3.0.24	3.0.24
python3-pycparser	2.20	2.20
python3-pyrsistent	0.17.3	0.17.3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022.7.1	2022.7.1
python3-pyyaml	5.4.1	5.4.1
python3-requests	2.25.1	2.25.1
python3-rpm	4.16.1.3	4.16.1.3
python3-ruamel-yaml	0.16.6	0.16.6
python3-ruamel-yaml-clib	0.1.2	0.1.2
python3-setools	4.4.1	4.4.1
python3-setuptools	59.6.0	59.6.0
python3-setuptools-wheel	59.6.0	59.6.0
python3-six	1.15.0	1.15.0
python3-systemd	235	235

Pacote	AMI	Hyper-V VHDX
python3-urllib3	1.25.10	1.25.10
python3-wcwidth	0.2.5	0.2.5
python-chevron	0.13.1	
python-srpm-macros	3.9	3.9
quota	4.06	4.06
quota-nls	4.06	4.06
readline	8.1	8.1
rng-tools	6.14	6.14
rootfiles	8.1	8.1
rpcbind	1.2.6	1.2.6
rpm	4.16.1.3	4.16.1.3
rpm-build-libs	4.16.1.3	4.16.1.3
rpm-libs	4.16.1.3	4.16.1.3
rpm-plugin-selinux	4.16.1.3	4.16.1.3
rpm-plugin-systemd-inhibit	4.16.1.3	4.16.1.3
rpm-sign-libs	4.16.1.3	4.16.1.3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0

Pacote	AMI	Hyper-V VHDX
sed	4.8	4.8
selinux-policy	37,22	37,22
selinux-policy-targeted	37,22	37,22
setup	2.13.7	2.13.7
shadow-utils	4,9	4,9
slang	2.3.2	2.3.2
sqlite-libs	3.40,0	3.40,0
sssd-client	2.9.4	2.9.4
sssd-common	2.9.4	
sssd-kcm	2.9.4	
sssd-nfs-idmap	2.9.4	
strace	5.16	5.16
sudo	1.9.14	1.9.14
sysctl-defaults	1.0	1,0
sysstat	12.5.6	12.5.6
systemd	252,16	252,16
systemd-libs	252,16	252,16
systemd-networkd	252,16	252,16
systemd-pam	252,16	252,16
systemd-resolved	252,16	252,16

Pacote	AMI	Hyper-V VHDX
systemd-udev	252,16	252,16
system-release	2023.4.20240319	2023.4.20240319
systemtap-runtime	4.8	4.8
tar	1,34	1,34
tbb	2020.3	2020.3
tcpdump	4.99.1	4.99.1
tcsh	24.6.07	24.6.07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2.37.4	2.37.4
util-linux-core	2.37.4	2.37.4
vim-common	9.0.2153	9.0.2153
vim-data	9.0.2153	9.0.2153
vim-enhanced	9.0.2153	9.0.2153
vim-filesystem	9.0.2153	9.0.2153
vim-minimal	9.0.2153	9.0.2153

Pacote	AMI	Hyper-V VHDX
wget	1.21.3	1.21.3
which	2.21	2.21
words	3.0	3.0
xfsdump	3.1.11	3.1.11
xfsprogs	5.18.0	5.18.0
xxd	9.0.2153	9.0.2153
xxhash-libs	0.8.0	0.8.0
xz	5.2.5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1.2.11	1.2.11
zram-generator	1.1.2	
zram-generator-defaults	1.1.2	
zstd	1.5.5	1.5.5

Layout do sistema de arquivos

Esta seção aborda o layout do sistema de arquivos de um sistema AL2 023, incluindo detalhes que podem ser específicos para instâncias ou AL2 contêineres baseados em 023. Veja o `file-hierarchy(7)` man página para obter mais informações.

Tópicos

/(O diretório raiz)

Por padrão, AL2 023 imagens são configuradas com um gravável/, permitindo que usuários privilegiados criem novos arquivos e diretórios.

É possível configurar `systemd` serviços para usar um caminho ou imagem diferente para aparecer como / para esse serviço, bem como colocar restrições de acesso em qualquer caminho.

> **Note**
>
> É uma prática recomendada que `systemd` os serviços sejam configurados para restringir o acesso ao qual o serviço tem. Isso pode incluir o uso da `ReadOnlyPaths=/` diretiva que torna a / leitura somente para esse serviço.
> Para obter mais informações sobre `systemd` como restringir o acesso que um serviço tem ao sistema, consulte o `systemd.exec(5)` man página.

`/boot`(Núcleo, initramfs, etc.)

Por padrão, as imagens AL2 023 inicializáveis são configuradas como `/boot` estando no root sistema de arquivos. O `/boot` caminho só é relevante para imagens inicializáveis, portanto, não é usado em imagens de contêiner AL2 023.

Esse diretório abriga os arquivos necessários para a inicialização do AL2 0.23, como o kernel Linux e initramfs. O conteúdo desse diretório só deve ser manipulado usando as ferramentas fornecidas com o sistema operacional.

`/boot/efi`(Partição do sistema EFI)

Por padrão, as imagens AL2 023 inicializáveis são configuradas com o EFI System partição sendo montada em`/boot/efi`. Esse sistema de arquivos é gerenciado pelo sistema operacional e contém código e configuração essenciais para inicializar o sistema.

Esse caminho não é relevante para imagens de contêiner.

`/etc`(Configuração do sistema)

O `/etc` diretório em AL2 023 contém a configuração específica do sistema. Por padrão, AL2 023 imagens vêm `/etc` com root sistema de arquivos e gravável por usuários privilegiados.

> **Note**
>
> É comum que os aplicativos (inclusive`systemd`) mantenham a configuração padrão sob a `/usr`(Recursos do sistema) qual pode ser substituída inserindo a configuração em. `/etc`(Configuração do sistema)
> Para esses aplicativos, alterar os arquivos em `/usr`(Recursos do sistema) vez de substituir a configuração padrão provavelmente `/etc` resultará na substituição das alterações quando o pacote for atualizado.

`/home`(Diretórios pessoais do usuário)

Usuários normais têm seus diretórios pessoais abaixo`/home`, mas o software deve sempre procurar a variável de `$HOME` ambiente por usuário, em vez de confiar em um padrão como. `/home/$USER`

Por padrão, AL2 023 imagens têm `/home` no root sistema de arquivos, mas o software não deve confiar nisso. É perfeitamente válido que o sistema operacional seja configurado `/home` para ser um sistema de arquivos separado, que é montado posteriormente durante a inicialização ou somente após a autenticação do usuário no sistema.

A ferramenta root o diretório inicial do usuário não está dentro, `/home` mas sim `/root` (root diretório inicial do usuário) para que esteja disponível caso o sistema de `/home` arquivos não possa ser montado.

> **Note**
>
> É a melhor prática para `systemd` serviços que não precisam de acesso de gravação `/home` serem configurados com a `ProtectHome=read-only` diretiva. Com essa opção,, `/home/root`, e `/run/user` são tornados somente para leitura para o serviço.
> Também é uma prática recomendada para serviços que não precisam de nenhum acesso `/home` para serem configurados com a `ProtectHome=tmpfs` diretiva, que executará o serviço em uma sandbox onde`/home`,`/root`, e `/run/user` são sistemas de arquivos vazios somente para leitura`tmpfs`.
> Para obter mais informações sobre `systemd` como restringir o acesso que um serviço tem ao sistema, consulte o `systemd.exec(5)` man página.

`/root` (root diretório inicial do usuário)

O diretório inicial do root user é o `/root` diretório, propositalmente separado /home(Diretórios pessoais do usuário) para que esteja presente no caso de estar em um sistema de arquivos que não está disponível. /home(Diretórios pessoais do usuário)

A melhor prática para configurar `systemd` serviços é a `/root` mesma para “for/home(Diretórios pessoais do usuário)”.

`/srv`(Carga útil do servidor)

O `/srv` diretório é gerenciado pelo administrador do sistema e o Amazon Linux 2023 não impõe restrições sobre como esse diretório é organizado.

É possível configurar o `/srv` diretório para estar em um sistema de arquivos separado, para que ele só fique disponível posteriormente na inicialização.

`/tmp`(pequenos arquivos temporários)

> **Note**
>
> O Amazon Linux 2023 é diferente do Amazon Linux 2, pois, por padrão, agora `/tmp` é `tmpfs` em vez de um caminho no root sistema de arquivos.

> **Note**
>
> Quando executado em um contêiner, normalmente é a configuração de tempo de execução do contêiner que determina se `/tmp` existe `tmpfs` ou um caminho no disco e se há um processo de limpeza em execução ou não.

O `/tmp` diretório é para arquivos temporários pequenos e limitados por tamanho. Por padrão, o AL2 023 o configura para ser um sistema de `tmpfs` arquivos com um limite de tamanho de 50% da RAM e um máximo de um milhão inodes.

Os aplicativos devem preferir o caminho na variável de `$TMPDIR` ambiente`/tmp`. Os usuários podem então definir a variável de `$TMPDIR` ambiente para substituir o caminho que um aplicativo deve usar para `/tmp`

Para arquivos temporários maiores, `/var/tmp` deve ser usado em vez disso.

> **Warning**
>
> Como `/tmp` é compartilhado, é importante usar métodos seguros para criar arquivos temporários. Para obter detalhes, consulte a `systemd` documentação inicial sobre Como usar `/tmp` e `/var/tmp` com segurança.

> **Note**
>
> É uma prática recomendada que os `systemd` serviços sejam configurados com a `PrivateTmp=` diretiva definida como `yes` ou `disconnected` que executa o serviço em uma sandbox onde `/tmp` e não `/var/tmp` sejam compartilhados com o host ou outros serviços.

Para obter mais informações, incluindo como configurar dois serviços para compartilhar os mesmos diretórios temporários privados, consulte o `systemd.exec(5)` man página.

Normalmente, o conteúdo do `/tmp` é limpo no momento da inicialização e os arquivos não utilizados são limpos regularmente. Por padrão, o processo de limpeza é executado logo após a inicialização e, em seguida, todos os dias. Para obter informações sobre como configurar a limpeza de arquivos temporários, consulte e `tmpfiles.d(5)` `systemd-tmpfiles(8)` man páginas principais.

Os `/var/tmp` caminhos `/tmp` e estão intimamente relacionados e existem para propósitos diferentes.

`/run`(dados de tempo de execução)

O `/run` diretório é usado pelos pacotes do sistema para armazenar pequenas quantidades de dados de tempo de execução (como arquivos de soquete). É um sistema de `tmpfs` arquivos e só pode ser gravado por programas privilegiados.

O `/run/log` diretório pode ser usado pelos componentes do sistema para armazenar os registros, antes de serem gravados `/var/log` ou antes que o sistema de `/var/log` arquivos esteja disponível.

O `/run/user/` caminho contém diretórios de tempo de execução por usuário. Por padrão, esses serão sistemas de `tmpfs` arquivos individuais montados `systemd` quando o usuário fizer login e apagados quando o usuário não estiver mais conectado. De acordo com a Especificação do Diretório Base do XDG, esses caminhos não devem ser referenciados diretamente, mas sim por meio da `$XDG_RUNTIME_DIR` variável de ambiente.

`/usr`(Recursos do sistema)

A `/usr` hierarquia é para recursos do sistema operacional fornecidos pelo fornecedor. Exceto pela `/usr/local` hierarquia, nada deve modificar nada, `/usr` exceto o gerenciador de pacotes do sistema operacional.

Os aplicativos de software devem presumir que isso `/usr` pode ser somente para leitura. A `/usr` hierarquia não deve ser usada para dados voláteis. Exceto que a `/usr` hierarquia não deve ser usada para nenhum dado adicionado ou alterado fora da instalação/remoção do pacote, conforme

feito pelo gerenciador de pacotes do sistema operacional. `/usr/local` O gerenciador de pacotes do sistema operacional pode presumir que toda a `/usr` hierarquia (exceto`/usr/local`) é o mesmo ponto de montagem.

O software que está sendo instalado fora do gerenciador de pacotes do sistema operacional não deve armazenar dados, `/usr` pois isso pode impedir qualquer invocação futura do gerenciador de pacotes do sistema operacional. A `/usr/local` hierarquia é a exceção e é reservada para software fora do gerenciador de pacotes do sistema operacional.

`/usr/bin`(Executáveis)

Arquivos executáveis que devem aparecer na pesquisa `$PATH` padrão e são úteis para invocar a partir de um shell. Daemons e executáveis que não são úteis para invocar a partir de um shell, em vez disso, residem em ou. `/usr/lib` `/usr/libexec`

`/usr/include`(Cabeçalhos C/C++)

O `/usr/include` diretório contém arquivos de cabeçalho C e C++, geralmente contidos em pacotes com o `-devel` sufixo.

`/usr/lib`e **`/usr/lib64`** (bibliotecas compartilhadas)

No Amazon Linux 2023, o `/usr/lib64` caminho é usado para bibliotecas compartilhadas de 64 bits e dados de pacotes que dependem da arquitetura. Como o AL2 023 não vem com nenhum suporte de espaço de usuário de 32 bits, há apenas bibliotecas compartilhadas de 64 bits disponíveis.

O `/usr/lib` caminho é para dados estáticos de pacotes de sistema operacional que são compatíveis com todas as arquiteturas. Isso pode incluir executáveis que geralmente não são invocados de um shell, que também podem ser encontrados em. `/usr/libexec` As bibliotecas compartilhadas são encontradas em `/usr/lib64` vez de`/usr/lib`.

`/usr/local`(Software instalado pelo administrador do sistema)

No Amazon Linux 2023, o `/usr/local` caminho está disponível para o administrador do sistema instalar software, software que não é de propriedade do sistema operacional e não será afetado pelo sistema operacional. A `/usr/local` hierarquia padrão reflete a hierarquia. `/`

`/usr/share`(Recursos compartilhados)

Recursos compartilhados, como documentação, fontes e dados de fuso horário, estão presentes/`usr/share`. É comum que várias especificações ditem exatamente onde e em qual formato os dados são armazenados nesse diretório.

`/usr/share/doc`(Documentação)

A documentação que vem com os pacotes será armazenada em`/usr/share/doc`.

`/var`(Dados variáveis persistentes do sistema)

`/var/cache`(Cache)

Por outro lado`/var/lib`, apagar dados não `/var/cache` resultará em perda de dados, pois os aplicativos precisam ser capazes de reconstruir seus `/var/cache` dados de outras fontes.

`/var/lib`(Dados persistentes do sistema)

O `/var/lib` diretório é usado para dados persistentes do sistema. Vários componentes do sistema colocarão aqui dados que são privados desse componente. Por outro lado`/var/cache`, apagar dados `/var/lib` resultará em perda de dados.

Por exemplo, o servidor de banco de dados PostgreSQL armazenará, por padrão, os dados do banco de dados em. `/var/lib/pgsql` O layout e os formatos de arquivo desses dados são privados do PostgreSQL e, se forem dados persistentes, pois se apagados, o usuário experimenta perda de dados.

`/var/log`(Registros persistentes)

Esse diretório é usado para armazenar registros persistentes. É recomendável que o software use as chamadas de `sd_journal_print(3)` API `syslog(3)` ou em vez de armazenar diretamente os arquivos de log em`/var/log`.

> **Note**
>
> Em AL2 023`systemd`diário substitui `rsyslog`, o que é uma diferença notável da configuração padrão do Amazon Linux 2.

Para obter mais informações sobre o uso de registros de leitura`journalctl`, consulte a página do `journalctl`manual.

Muitos aplicativos usam seus próprios mecanismos para gravar e, às vezes, girar os arquivos de log encontrados em`/var/log`. Consulte a documentação desses aplicativos sobre como configurar seus arquivos de log.

`/var/spool`(Filas de correio e impressora)

Esse diretório é usado para dados persistentes, como filas de correio ou impressoras.

`/var/tmp`(arquivos temporários maiores)

Para arquivos temporários pequenos e limitados por tamanho, possivelmente `/tmp` deve ser usado em vez disso.

Embora `/tmp` esteja configurado por padrão para ser um `tmpfs` volume, por padrão `/var/tmp` é configurado para ser um caminho no sistema de arquivos raiz e, portanto, é o local para arquivos temporários maiores e mais persistentes. Por padrão, há um trabalho de limpeza executado em uma programação regular que remove arquivos não acessados recentemente.

Para obter informações sobre como configurar a limpeza de arquivos temporários, consulte e `tmpfiles.d(5)` `systemd-tmpfiles(8)` man páginas principais.

Da mesma forma`/tmp`, os aplicativos devem preferir o caminho especificado na variável de `$TMPDIR` ambiente`/var/tmp`. Os usuários podem então definir a variável de `$TMPDIR` ambiente para substituir o caminho para `/var/tmp` o qual um aplicativo deve usar.

> **Warning**
>
> Como `/var/tmp` é compartilhado (como está)`/tmp`, é importante usar métodos seguros para criar arquivos temporários. Para obter detalhes, consulte a `systemd` documentação inicial sobre Como usar `/tmp` e `/var/tmp` com segurança.

> **Note**
>
> É uma prática recomendada que os `systemd` serviços sejam configurados com a `PrivateTmp=` diretiva definida como `yes` ou `disconnected` que executa o serviço em uma sandbox onde `/tmp` e não `/var/tmp` sejam compartilhados com o host ou outros serviços.

Para obter mais informações, incluindo como configurar dois serviços para compartilhar os mesmos diretórios temporários privados, consulte o `systemd.exec(5)` man página.

Os `/var/tmp` caminhos `/tmp` e estão intimamente relacionados e existem para propósitos diferentes.

Atualizando AL2 023

É importante manter-se atualizado com as versões AL2 023 para que você possa se beneficiar das atualizações de segurança e dos novos recursos. Com o AL2 023, você pode garantir a consistência entre as versões e atualizações do pacote em seu ambiente por meio Atualizações determinísticas por meio de repositórios versionados em 023 AL2 de.

> **Warning**
>
> `dnf --releasever=latest update`A execução não é uma prática recomendada e provavelmente resultará no primeiro teste de uma atualização do sistema operacional em produção.
> Em vez de usar`latest`, use uma versão específica da versão AL2 023. Isso garante que você esteja implantando as mesmas alterações em todas as instâncias de produção que você testou anteriormente. Por exemplo, sempre `dnf --releasever=2023.7.20250331 update` atualizará para a versão 2023.7.20250331.
> Para obter mais informações, consulte a seção Atualizando o AL2 023 no Guia do usuário do AL2 023.

Tópicos

Práticas recomendadas para implantar atualizações com segurança

O Amazon Linux 2023 (AL2023) tem vários recursos projetados para ajudar na implantação segura de atualizações no sistema operacional e na capacidade de saber o que mudou entre as

atualizações e, se necessário, reverter facilmente para a versão mais antiga. Esta seção explora as lições aprendidas em mais AWS de uma década de uso interno e externo do Amazon Linux.

> **Warning**
>
> `dnf --releasever=latest update`A execução não é uma prática recomendada e provavelmente resultará em uma atualização do sistema operacional sendo testada pela primeira vez em produção.
> Em vez de usar`latest`, use uma versão específica da versão AL2 023. Isso garante que você esteja implantando as mesmas alterações em todas as instâncias de produção que você testou anteriormente. Por exemplo, sempre `dnf --releasever=2023.7.20250331 update` atualizará para a versão 2023.7.20250331.
> Para obter mais informações, consulte a seção Atualizando o AL2 023 no Guia do usuário do AL2 023.

Sem planejar a segurança da implantação das atualizações do sistema operacional, o impacto de uma interação negativa inesperada entre seu aplicativo/serviço e uma atualização do sistema operacional pode ser significativamente maior, incluindo uma interrupção total. Como acontece com qualquer problema de software, quanto mais cedo o problema for detectado, menor será o impacto que ele poderá ter sobre os usuários finais.

É importante não cair na armadilha de acreditar em duas coisas que fundamentalmente não são verdadeiras:

1. O fornecedor do sistema operacional nunca cometerá um erro ao atualizar o sistema operacional.
2. O comportamento específico ou a interface do sistema operacional em que você confia corresponde ao comportamento e às interfaces nas quais o fornecedor do sistema operacional consideraria algo confiável.

   ou seja, tanto o fornecedor do sistema operacional quanto você concordariam que houve um problema com a atualização.

Não confie em boas intenções, implemente sistemas para garantir que a segurança da implantação inclua qualquer atualização do sistema operacional.

Não é recomendável testar novas atualizações do sistema operacional implantando-as em ambientes de produção. É uma prática recomendada considerar o sistema operacional como outra

parte de sua implantação e pensar em aplicar os mesmos mecanismos de segurança de implantação que você considera adequados para qualquer outra alteração em um ambiente de produção.

É uma prática recomendada testar toda e qualquer atualização do sistema operacional antes da implantação nos sistemas de produção. Durante a implantação, são recomendadas implementações graduais combinadas com um bom monitoramento. As implementações graduais podem garantir que, se um problema ocorrer, mesmo que não seja imediato, o impacto seja restrito a um subconjunto de uma frota, e a implantação posterior da atualização possa ser interrompida enquanto outras investigações e mitigações podem ocorrer.

A mitigação de qualquer impacto negativo da atualização do sistema operacional geralmente é a primeira prioridade, seguida pela resolução do problema, onde quer que ele esteja. Quando a introdução de uma atualização do sistema operacional está relacionada a um impacto negativo, a capacidade de reverter para a versão anterior em boas condições do sistema operacional é uma ferramenta poderosa.

O Amazon Linux 2023 apresenta Atualizações determinísticas por meio de repositórios versionados um novo recurso poderoso para garantir que qualquer alteração na versão do sistema operacional (ou pacotes individuais) seja repetível. Portanto, se for encontrado um problema ao passar de uma versão do sistema operacional para a próxima, existem mecanismos simples de usar disponíveis para manter a versão do sistema operacional que funciona e, ao mesmo tempo, descobrir como resolver o problema.

Com o AL2 023, sempre que lançamos novas atualizações de pacotes, há uma nova versão para bloquear e um novo AMIs bloqueio para essa versão. As notas de lançamento da versão AL2 023 abordam as mudanças em cada versão e Consultorias de segurança do Amazon Linux para 2013 AL2 abordam os problemas de segurança abordados nas atualizações do pacote.

Por exemplo, se você foi afetado pelo problema presente na versão 2023.6.20241028, você poderia imediatamente voltar a usar as imagens de contêiner da versão anterior, 2023.6.20241010. AMIs Nesse caso, havia um bug em um pacote que foi corrigido na versão 2023.6.20241031 subsequente, mas com Atualizações determinísticas por meio de repositórios versionados qualquer pessoa afetada, poderia imediatamente tomar uma ação simples para mitigar: basta usar as imagens anteriores.

Atualizações determinísticas por meio de repositórios versionadostambém garante que qualquer implantação em andamento de uma atualização do sistema operacional, seja no local ou por meio do lançamento de imagens novas AMIs ou de contêiner, não seja afetada pelas atualizações do sistema operacional lançadas posteriormente.

Para nosso primeiro exemplo, a frota A é uma grande frota que está na metade da implantação da atualização de 2023.5.20241001 para a versão 2023.6.20241010 quando a versão 2023.6.20241028 for lançada. Atualizações determinísticas por meio de repositórios versionadossignifica que a implantação da frota A continua sem nenhuma alteração nas atualizações que ela está aplicando.

O objetivo das estratégias de implantação baseadas em ondas ou em fases, como implantar primeiro em 1% de uma frota, depois 5%, 10%, 20%, 40%, até atingir 100%, é poder testar uma mudança de forma limitada antes de implementá-la mais amplamente. Esse tipo de estratégia de implantação geralmente é considerado a melhor prática para implantar qualquer alteração na produção.

Com uma estratégia de implantação baseada em ondas e a frota. Uma atualização para 2023.6.20241010 está em um estágio em que está sendo implantada em vários hosts ao mesmo tempo, o fato de 2023.6.20241028 ter sido lançada não afeta a implantação em andamento graças ao uso. Atualizações determinísticas por meio de repositórios versionados

Se a frota B estivesse executando uma versão mais antiga, digamos, 2023.5.20240708, e tivesse começado a implantar a atualização para 2023.6.20241028, e a frota B fosse afetada pelo problema nessa versão, isso seria notado logo no início da implantação. Nesse ponto, pode-se decidir se deve pausar qualquer implementação até que uma correção para esse problema esteja disponível ou se, entretanto, iniciar uma implantação da mesma versão que a frota A estava em execução, 2023.6.20241010 para que a frota B receba todas as atualizações entre 2023.5.20240708 e 2023.6.20241010.

É importante observar que não receber atualizações do sistema operacional imediatamente pode causar problemas. As novas atualizações provavelmente contêm correções de bugs e segurança que podem ser relevantes para o seu ambiente. Para obter mais informações, consulte Segurança e compatibilidade no Amazon Linux 2023 e Gerencie atualizações de pacotes e sistemas operacionais em AL2 023.

É importante configurar seus sistemas de implantação para poder receber facilmente novas atualizações do sistema operacional, testá-las antes da implantação na produção e usar mecanismos como implantações baseadas em ondas para minimizar qualquer impacto negativo. Para poder mitigar qualquer impacto negativo de uma atualização do sistema operacional, é importante saber como fazer com que seus sistemas de implantação apontem para uma versão anterior em boas condições do sistema operacional e, depois que o problema for resolvido, não fique mais preso à versão antiga em boas condições, mas sim migrar para uma nova versão em boas condições.

Preparando-se para pequenas atualizações

A preparação para atualizações menores do sistema operacional, como uma nova versão pontual do AL2 023, deve ser limitada a zero esforço. Não deixe de ler as notas de lançamento da versão AL2 023 para ver as próximas alterações.

O período de suporte de um pacote chegando ao fim pode envolver a mudança para uma versão mais recente do tempo de execução da linguagem (como withPHP em AL2 023). É uma prática recomendada se preparar para isso com antecedência, migrando para versões de tempo de execução em novos idiomas confortavelmente antes do término do período de suporte.

Para pacotes como`pcre` versão 1, também há a oportunidade de planejar com antecedência e migrar qualquer código para seu substituto, que neste caso é a `pcre` versão 2. É uma boa prática fazer isso o mais rápido possível, para ter tempo para qualquer contratempo.

Onde não há substituição direta, como comBerkeley DB (2) `libdb`, talvez seja necessário fazer uma escolha com base no seu caso de uso.

Preparando-se para atualizações importantes

A atualização para uma nova versão principal de um sistema operacional é quase universalmente vista como algo que requer planejamento, trabalho para se adaptar a funcionalidades alteradas ou obsoletas e também testes antes da implantação. Não é incomum poder se preparar para a próxima versão principal do Amazon Linux 2023 de forma mais incremental, como abordar qualquer uso de funcionalidades obsoletas ou removidas antes de prosseguir com a mudança para a próxima versão principal.

Por exemplo, ao passar de AL2 para AL2 023, a leitura da Funcionalidade obsoleta AL2 e removida em 023 AL2 seção pode resultar em várias etapas pequenas e seguras que podem ocorrer enquanto ainda é usada para se preparar AL2 para AL2 023. Por exemplo, qualquer O Python 2.7 foi substituído pelo Python 3 uso (fora do uso do sistema operacional, como no gerenciador de `yum` pacotes) pode ser migrado para o Python 3 em preparação para o uso. Python em AL2 023 Se estiver usando PHP, tanto AL2 (por meio do PHP 8.2 AL2 Extra) quanto o AL2 023 fornecem o PHP 8.2 e, portanto, a migração da versão do PHP e a migração do sistema operacional não precisam ocorrer simultaneamente.

Ao usar o AL2 023, também é possível se preparar para a próxima versão principal do Amazon Linux 2023 hoje, usando o AL2 023. A Obsoleto em 2023 AL2 seção aborda recursos e pacotes que estão obsoletos na versão AL2 023 e que devem ser removidos.

Por exemplo, migrar qualquer System V init (`sysvinit`) uso restante, como `init` scripts, para seus `systemd` equivalentes preparará você para o futuro, além de permitir que você use o conjunto completo de `systemd` recursos para monitorar o serviço, como e se reiniciá-lo, quais outros serviços ele precisa e se alguma restrição de recurso ou permissão deve ser aplicada.

Para recursos como suporte de 32 bits, a suspensão de uso pode abranger várias versões principais do sistema operacional. Para 32 bits, o Amazon Linux 1 (AL1) está obsoletox86 de 32 bits (i686) AMIs, o Amazon Linux 2 está obsoleto e o Amazon Linux 2023 está Pacotes x86 (i686) de 32 bits obsoleto. Suporte de tempo de execução x86 (i686) de 32 bits A transição do IMDSv1 também abrange várias versões principais do sistema operacional. Para esses tipos de mudanças, entende-se que alguns clientes precisam de mais tempo para se adaptar a elas, portanto, há uma grande margem de manobra antes que a funcionalidade não esteja mais disponível no Amazon Linux 2023.

A lista de funcionalidades obsoletas é atualizada durante a vida útil do sistema operacional, e é recomendável manter-se atualizado com as alterações feitas nela.

Receba notificações sobre novas atualizações

Você pode receber notificações sempre que uma nova AMI AL2 023 for lançada. As notificações são publicadas com o Amazon SNS usando o tópico a seguir.

```
arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates
```

As mensagens são publicadas aqui quando uma nova AMI AL2 023 é publicada. A versão da AMI será incluída na mensagem.

Essas mensagens podem ser recebidas usando vários métodos diferentes. Recomendamos que você use o método a seguir.

1. Abra o console do Amazon SNS.
2. Na barra de navegação, altere Região da AWS para Leste dos EUA (Norte da Virgínia), se necessário. É necessário selecionar esta região onde a notificação do SNS que está assinando foi criada nesta região.
3. No painel de navegação, escolha Assinaturas, Criar assinatura.
4. Na caixa de diálogo Create subscription, faça o seguinte:
   a. Para o ARN do tópico, copie e cole o seguinte nome de recurso da Amazon (ARN):. **`arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates`**
   b. Em Protocol (Protocolo), escolha Email.

c. Em Endpoint, insira um endereço de e-mail que possa ser usado para receber notificações.

d. Selecione Create subscription.

5. Você recebe um e-mail de confirmação com o assunto "AWS Notificação - Confirmação de assinatura". Abra o e-mail e escolha Confirm subscription para concluir a assinatura.

Atualizações determinísticas por meio de repositórios versionados em 023 AL2

> **Note**
>
> Por padrão, sua instância AL2 023 não recebe automaticamente atualizações de segurança adicionais críticas e importantes na inicialização. Sua instância contém inicialmente as atualizações que estavam disponíveis na versão AL2 023 e na AMI escolhida.

Controle as atualizações recebidas de versões principais e secundárias

Com o AL2 023, você pode garantir a consistência entre as versões e atualizações do pacote em seu ambiente. Você também pode garantir a consistência de várias instâncias da mesma Imagem de máquina da Amazon (AMI). Com as atualizações determinísticas através do recurso de repositórios de versão, que é ativado por padrão, você pode aplicar atualizações com base em um cronograma que atenda às suas necessidades específicas.

Sempre que lançamos novas atualizações de pacotes, há uma nova versão para bloquear e um novo AMIs bloqueio para essa versão.

AL2023 bloqueia uma versão específica do seu repositório. Isso é compatível com versões principais ou secundárias. A AMI AL2 023, exposta por meio de nossos parâmetros SSM, é sempre a versão mais recente. Ele tem a maioria dos up-to-date pacotes e atualizações, incluindo atualizações de segurança críticas e importantes.

Se você iniciar uma instância em uma AMI existente, as atualizações não são aplicadas automaticamente. Todos os pacotes adicionais instalados como parte do seu aprovisionamento são mapeados para a versão do repositório da AMI existente.

Com esse recurso, você é responsável por garantir a consistência entre as versões e atualizações do pacote em seu ambiente. Esse é particularmente o caso se você estiver executando várias instâncias

da mesma AMI. É possível aplicar atualizações baseadas em um cronograma que atenda às suas necessidades. Você também pode aplicar um conjunto específico de atualizações no lançamento, pois elas também podem ser bloqueadas em uma versão específica do repositório.

Diferenças entre atualizações de versão menor e principal

As versões principais do AL2 023 incluem atualizações em grande escala e podem adicionar, excluir ou atualizar pacotes. Para garantir a compatibilidade, atualize sua instância para uma nova versão principal somente depois de testar seu aplicativo nessa versão.

Versões secundárias do AL2 023 incluem atualizações de recursos e segurança, mas não incluem alterações de pacotes. Isso garante que os recursos do Linux e a API da biblioteca do sistema permaneçam disponíveis em novas versões. Não é necessário testar o aplicativo antes da atualização.

Saber quando as atualizações estão disponíveis

Para aplicar uma atualização, você precisa saber se há uma disponível e saber como implantá-la.

Para compilações derivadas AMIs quando novos AL2 023 AMIs são lançados, o EC2 Image Builder pode criar, corrigir e testar AMIs automaticamente. Para acionar seus próprios pipelines de construção de AMI ou usar a base AMIs, você podeReceba notificações sobre novas atualizações.

Para aplicar patches no local, você pode usar ferramentas como o AWS Systems Manager Patch Manager para orquestrar a aplicação de atualizações em uma frota.

Para outros públicos AMIs com base no AL2 023, os fornecedores desses AMIs podem ter seu próprio cronograma de lançamentos e métodos de notificação. Ao usar imagens derivadas AMIs ou de contêiner, verifique a documentação do editor para saber quando as atualizações serão lançadas.

As mudanças em cada versão estão documentadas nas notas da versão AL2 023. As atualizações de segurança são publicadas no Amazon Linux Security Center (ALAS).

Controle as atualizações de pacotes disponíveis nos repositórios AL2 023

Quando publicamos uma nova versão dos repositórios AL2 023, todas as versões anteriores ainda estão disponíveis. Por padrão, o plug-in para gerenciar versões do repositório se fixa na mesma versão usada para criar a AMI. Se pretende controlar as atualizações do pacote, siga estas etapas.

1. Descubra as versões disponíveis do repositório ao executar o comando a seguir.

```
$ sudo dnf check-release-update
```

2. O comando a seguir pode ser executado para verificar.

```
$ sudo dnf upgrade --releasever=version
```

Esse comando inicia uma atualização usando `dnf` a partir da versão atual de lançamento Amazon Linux para a versão de lançamento que é especificada na linha de comando. Uma lista das atualizações do pacote é apresentada por `dnf`. Antes que a atualização seja processada, você deve confirmar a atualização. Depois que a atualização for concluída, a nova versão de lançamento se tornará a versão de lançamento padrão que `dnf` usa para todas as atividades futuras.

Para obter mais informações, consulte Gerencie atualizações de pacotes e sistemas operacionais em AL2 023.

Atualizações determinísticas por meio da substituição de instâncias

O Atualizações determinísticas por meio de repositórios versionados em 023 AL2 recurso do Amazon Linux 2023 torna a substituição de instâncias uma maneira fácil de implementar de forma determinística e segura as versões atualizadas do AL2 023. Atualizações determinísticas significam que, à medida que uma nova versão é lançada progressivamente, se algum problema for encontrado, é simples reverter para a AMI anterior e, ao mesmo tempo, determinar a causa do problema.

Usar a substituição de instâncias em vez de aplicar patches no local significa que as atualizações são mais determinísticas e previsíveis, pois o lançamento de novas capacidades pode ser um caminho de código bem testado com estados A e B claros. Cada um dos estados antes e depois pode ser bem testado em um sistema CI/CD antes do início da implantação.

Ao aplicar patches no local, há muitos estados intermediários entre antes e depois da aplicação das atualizações, o que é mais difícil de testar para todas as combinações de estados.

Uma estratégia de atualização do sistema operacional usando a substituição de instâncias com atualizações determinísticas se encaixa bem nos modelos de implantação azul/verde, em ondas e baseados em fases.

Usando atualizações determinísticas por meio de repositórios versionados

Tópicos

Usando um sistema determinístico atualizado

> **Note**
>
> O comportamento padrão do gerenciador de pacotes mudou de AL2.

As atualizações determinísticas são uma forma poderosa de garantir que todas as mudanças nos ambientes de produção possam ser totalmente testadas antes de uma ampla implantação. Cada nova AMI AL2 023 é bloqueada para uma versão específica de AL2 023. Isso fornece um comportamento determinístico de quais versões dos pacotes de sistema operacional são instaladas ao iniciar a AMI específica. As atualizações no local podem ser para uma versão de lançamento específica, garantindo um comportamento determinístico em toda a frota. Ao migrar para versões de atualização novas AMIs ou in-loco, você pode testar cada uma em seu pipeline de CI/CD, detectando possíveis problemas antes de implantá-las em ambientes de produção.

Você pode usar ferramentas como o AWS Systems Manager Patch Manager para orquestrar a aplicação de atualizações em uma frota. Para criar derivados AMIs quando o novo AL2 023 AMIs for lançado, o EC2 Image Builder pode criar, corrigir e testar automaticamente AMIs, ou você pode Receba notificações sobre novas atualizações saber quando uma nova base AMIs está disponível ou acionar seus próprios pipelines de criação de AMI.

Para obter informações sobre como restringir as atualizações às de um determinado comunicado, consulte Aplicando atualizações de segurança no local

Para aplicar patches no local, você pode usar o gerenciador de `dnf` pacotes. Quando você executa o comando `dnf upgrade`, o sistema verifica se há atualizações no repositório que a variável `releasever` especifica. Uma versão válida `releasever` é uma *`latest`* ou uma versão com carimbo de data, como. *`2023.7.20250331`*

É possível alterar o valor de `releasever` usando um dos métodos a seguir. Esses métodos estão listados em prioridade decrescente do sistema. Isso significa que o método 1 substitui os métodos 2 e 3, e o método 2 substitui o método 3.

1. O valor no sinalizador da linha de comando, `--releasever=`*`latest`*, se for usado.
2. O valor especificado no arquivo da variável de substituição, `/etc/dnf/vars/releasever`, se estiver definido.
3. A versão atualmente instalada do pacote `system-release`.

No exemplo a seguir, a versão é*`2023.0.20230210`*:

```
$ rpm -q system-release
system-release-2023.0.20230210-0.amzn2023.noarch
```

Em um sistema recém-instalado, a variável de substituição não está presente. Nenhuma atualização está disponível porque o sistema está bloqueado para a versão instalada do `system-release`.

```
$ cat /etc/dnf/vars/releasever
cat: /etc/dnf/vars/releasever: No such file or directory
```

```
$ sudo dnf upgrade
Last metadata expiration check: 0:00:02 ago on Wed 15 Feb 2023 06:14:12 PM UTC.
Dependencies resolved.
Nothing to do.
Complete!
```

Você pode obter pacotes de uma versão específica usando o sinalizador `releasever` para fornecer a versão desejada.

```
$ rpm -q system-release
system-release-2023.0.20230222-0.amzn2023.noarch
```

```
$ sudo dnf upgrade --releasever=2023.0.20230329
Amazon Linux 2023 repository                    26 MB/s |  12 MB     00:00
Dependencies resolved.
================================================================================
 Package                 Arch    Version                      Repository   Size
```

```
================================================================================
Installing:
 kernel                   aarch64 6.1.21-1.45.amzn2023        amazonlinux  26 M
Upgrading:
 amazon-linux-repo-s3     noarch  2023.0.20230329-0.amzn2023  amazonlinux  18 k
 ca-certificates          noarch  2023.2.60-1.0.amzn2023.0.1  amazonlinux 828 k
 cloud-init               noarch  22.2.2-1.amzn2023.1.7       amazonlinux 1.1 M

         ... [ list edited for clarity ]

 system-release           noarch  2023.0.20230329-0.amzn2023  amazonlinux  29 k

         ... [ list edited for clarity ]

 vim-data                 noarch  2:9.0.1403-1.amzn2023.0.1   amazonlinux  25 k
 vim-minimal              aarch64 2:9.0.1403-1.amzn2023.0.1   amazonlinux 753 k

Transaction Summary
================================================================================
Install    1 Package
Upgrade   42 Packages

Total download size: 56 M
```

Como a opção `--releasever` substitui ambas `system-release` e `/etc/dnf/vars/releasever`, o resultado dessa atualização é o seguinte:

1. A atualização substitui todos os pacotes instalados que foram alterados entre a versão anterior e a nova.
2. A atualização bloqueia o sistema no repositório da nova versão do `system-release`.

Ao sempre especificar para qual `releasever` (ou seja, versão AL2 023) atualizar, você tem um conjunto determinístico de mudanças em uma frota. Você lançou a versão*A*, atualizou para *B* e depois atualizou para*C*.

Atualização seletiva de um sistema determinístico atualizado

> **Note**
>
> Recomendamos que todas as atualizações em uma nova versão sejam instaladas em vez de selecionar atualizações específicas. Aplicar apenas parte de uma atualização ao sistema operacional deve ser uma exceção à prática padrão de fazer a atualização inteira.

Talvez você queira instalar pacotes selecionados de uma versão recente, deixando o sistema bloqueado para a versão original.

É possível usar `dnf check-update` para identificar os pacotes que você deseja atualizar.

```
$ sudo dnf check-update --releasever=latest --security
Amazon Linux 2023 repository                    13 MB/s |  10 MB     00:00
Last metadata expiration check: 0:00:02 ago on Wed 15 Feb 2023 02:52:21 AM UTC.

bind-libs.aarch64                 32:9.16.27-1.amzn2023.0.1         amazonlinux
bind-license.noarch               32:9.16.27-1.amzn2023.0.1         amazonlinux
bind-utils.aarch64                32:9.16.27-1.amzn2023.0.1         amazonlinux
cryptsetup.aarch64                2.4.3-2.amzn2023.0.1              amazonlinux
cryptsetup-libs.aarch64           2.4.3-2.amzn2023.0.1              amazonlinux
curl-minimal.aarch64              7.85.0-1.amzn2023.0.1             amazonlinux
glibc.aarch64                     2.34-40.amzn2023.0.2              amazonlinux
glibc-all-langpacks.aarch64       2.34-40.amzn2023.0.2              amazonlinux
glibc-common.aarch64              2.34-40.amzn2023.0.2              amazonlinux
glibc-locale-source.aarch64       2.34-40.amzn2023.0.2              amazonlinux
gmp.aarch64                       1:6.2.1-2.amzn2023.0.1            amazonlinux
gnupg2-minimal.aarch64            2.3.7-1.amzn2023.0.2              amazonlinux
gzip.aarch64                      1.10-5.amzn2023.0.1               amazonlinux
kernel.aarch64                    6.1.12-17.42.amzn2023             amazonlinux
kernel-tools.aarch64              6.1.12-17.42.amzn2023             amazonlinux
libarchive.aarch64                3.5.3-2.amzn2023.0.1              amazonlinux
libcurl-minimal.aarch64           7.85.0-1.amzn2023.0.1             amazonlinux
libsepol.aarch64                  3.4-3.amzn2023.0.2                amazonlinux
libsolv.aarch64                   0.7.22-1.amzn2023.0.1             amazonlinux
libxml2.aarch64                   2.9.14-1.amzn2023.0.1             amazonlinux
logrotate.aarch64                 3.20.1-2.amzn2023.0.2             amazonlinux
lua-libs.aarch64                  5.4.4-3.amzn2023.0.1              amazonlinux
lz4-libs.aarch64                  1.9.4-1.amzn2023.0.1              amazonlinux
openssl.aarch64                   1:3.0.5-1.amzn2023.0.3            amazonlinux
```

```
openssl-libs.aarch64                1:3.0.5-1.amzn2023.0.3          amazonlinux
pcre2.aarch64                       10.40-1.amzn2023.0.1            amazonlinux
pcre2-syntax.noarch                 10.40-1.amzn2023.0.1            amazonlinux
rsync.aarch64                       3.2.6-1.amzn2023.0.2            amazonlinux
vim-common.aarch64                  2:9.0.475-1.amzn2023.0.1        amazonlinux
vim-data.noarch                     2:9.0.475-1.amzn2023.0.1        amazonlinux
vim-enhanced.aarch64                2:9.0.475-1.amzn2023.0.1        amazonlinux
vim-filesystem.noarch               2:9.0.475-1.amzn2023.0.1        amazonlinux
vim-minimal.aarch64                 2:9.0.475-1.amzn2023.0.1        amazonlinux
xz.aarch64                          5.2.5-9.amzn2023.0.1            amazonlinux
xz-libs.aarch64                     5.2.5-9.amzn2023.0.1            amazonlinux
zlib.aarch64                        1.2.11-32.amzn2023.0.3          amazonlinux
```

Instale os pacotes que você deseja atualizar. Use `sudo dnf upgrade --releasever=latest` e os nomes dos pacotes para garantir que o pacote `system-release` permaneça inalterado.

```
$ sudo dnf upgrade --releasever=latest openssl openssl-libs
Last metadata expiration check: 0:01:28 ago on Wed 15 Feb 2023 02:52:21 AM UTC.
Dependencies resolved.
================================================================================
 Package          Arch        Version                       Repository       Size
================================================================================
Upgrading:
 openssl          aarch64     1:3.0.5-1.amzn2023.0.3        amazonlinux     1.1 M
 openssl-libs     aarch64     1:3.0.5-1.amzn2023.0.3        amazonlinux     2.1 M

Transaction Summary
================================================================================
Upgrade  2 Packages

Total download size: 3.2 M
```

> **Note**
>
> O uso de `sudo dnf upgrade --releasever=latest` atualiza todos os pacotes, inclusive `system-release`. Em seguida, a versão permanece bloqueada para o novo `system-release`, a menos que você defina a substituição persistente.

Usando a substituição persistente com atualização determinística

> **Note**
>
> Com atualizações determinísticas, você pode integrar as alterações do sistema operacional ao seu pipeline de CI/CD. A desativação das atualizações determinísticas remove a capacidade de testar antes da implantação.

Em vez de adicionar`--releasever=latest`, você pode usar a substituição persistente para desbloquear o sistema definindo o valor da variável como*`latest`*. Ao usar sempre`latest`, isso reverte o comportamento de AL2 023 para o modelo de AL2 atualização, em que qualquer chamada para o gerenciador de pacotes sempre examinará a versão mais recente e não está bloqueada em nenhuma versão específica do sistema operacional.

> **Warning**
>
> Ao desbloquear o gerenciador de pacotes usando uma substituição persistente de atualizações determinísticas, você corre o risco de descobrir qualquer possível incompatibilidade entre seu aplicativo e uma atualização do sistema operacional em produção.
>
> Embora as incompatibilidades sejam raras, com uma atualização do sistema operacional você está integrando novas alterações de código em seu ambiente, os testes de integração podem impedir a implantação de alterações de código que tenham um impacto negativo nos ambientes de produção.

```
$ echo latest | sudo tee /etc/dnf/vars/releasever
latest
```

```
$ sudo dnf upgrade
Last metadata expiration check: 0:03:36 ago on Wed 15 Feb 2023 02:52:21 AM UTC.
Dependencies resolved.
================================================================================
 Package                 Arch    Version                       Repository   Size
================================================================================
Installing:
 kernel                  aarch64 6.1.73-45.135.amzn2023        amazonlinux  24 M
```

```
Upgrading:
 acl                     aarch64 2.3.1-2.amzn2023.0.1         amazonlinux  72 k
 alternatives            aarch64 1.15-2.amzn2023.0.1          amazonlinux  36 k
 amazon-ec2-net-utils    noarch  2.3.0-1.amzn2023.0.1         amazonlinux  16 k
 at                      aarch64 3.1.23-6.amzn2023.0.1        amazonlinux  60 k
 attr                    aarch64 2.5.1-3.amzn2023.0.1         amazonlinux  59 k
 audit                   aarch64 3.0.6-1.amzn2023.0.1         amazonlinux 249 k
 audit-libs              aarch64 3.0.6-1.amzn2023.0.1         amazonlinux 116 k
 aws-c-auth-libs         aarch64 0.6.5-6.amzn2023.0.2         amazonlinux  79 k
 aws-c-cal-libs          aarch64 0.5.12-7.amzn2023.0.2        amazonlinux  34 k
 aws-c-common-libs       aarch64 0.6.14-6.amzn2023.0.2        amazonlinux 119 k
 aws-c-compression-libs  aarch64 0.2.14-5.amzn2023.0.2        amazonlinux  22 k
 aws-c-event-stream-libs aarch64 0.2.7-5.amzn2023.0.2         amazonlinux  47 k
 aws-c-http-libs         aarch64 0.6.8-6.amzn2023.0.2         amazonlinux 147 k
 aws-c-io-libs           aarch64 0.10.12-5.amzn2023.0.6       amazonlinux 109 k
 aws-c-mqtt-libs         aarch64 0.7.8-7.amzn2023.0.2         amazonlinux  61 k
 aws-c-s3-libs           aarch64 0.1.27-5.amzn2023.0.3        amazonlinux  54 k
 aws-c-sdkutils-libs     aarch64 0.1.1-5.amzn2023.0.2         amazonlinux  26 k
 aws-checksums-libs      aarch64 0.1.12-5.amzn2023.0.2        amazonlinux  50 k
 awscli-2                noarch  2.7.8-1.amzn2023.0.4         amazonlinux 7.3 M
 basesystem              noarch  11-11.amzn2023.0.1           amazonlinux 7.8 k
 bash                    aarch64 5.1.8-2.amzn2023.0.1         amazonlinux 1.6 M
 bash-completion         noarch  1:2.11-2.amzn2023.0.1        amazonlinux 292 k
 bc                      aarch64 1.07.1-14.amzn2023.0.1       amazonlinux 120 k
 bind-libs               aarch64 32:9.16.27-1.amzn2023.0.1    amazonlinux 1.2 M
 bind-license            noarch  32:9.16.27-1.amzn2023.0.1    amazonlinux  14 k
 bind-utils              aarch64 32:9.16.27-1.amzn2023.0.1    amazonlinux 206 k
 binutils                aarch64 2.38-20.amzn2023.0.3         amazonlinux 4.6 M
 boost-filesystem        aarch64 1.75.0-4.amzn2023.0.1        amazonlinux  55 k
 boost-system            aarch64 1.75.0-4.amzn2023.0.1        amazonlinux  14 k
 boost-thread            aarch64 1.75.0-4.amzn2023.0.1        amazonlinux  54 k
 bzip2                   aarch64 1.0.8-6.amzn2023.0.1         amazonlinux  53 k
 bzip2-libs              aarch64 1.0.8-6.amzn2023.0.1         amazonlinux  44 k
 c-ares                  aarch64 1.17.2-1.amzn2023.0.1        amazonlinux 107 k
 ca-certificates         noarch  2021.2.50-1.0.amzn2023.0.3   amazonlinux 343 k
 checkpolicy             aarch64 3.4-3.amzn2023.0.1           amazonlinux 345 k
 chkconfig               aarch64 1.15-2.amzn2023.0.1          amazonlinux 162 k
 chrony                  aarch64 4.2-7.amzn2023.0.4           amazonlinux 314 k
 cloud-init              noarch  22.2.2-1.amzn2023.1.7        amazonlinux 1.1 M
 cloud-utils-growpart    aarch64 0.31-8.amzn2023.0.2          amazonlinux  31 k
 coreutils               aarch64 8.32-30.amzn2023.0.2         amazonlinux 1.1 M
 coreutils-common        aarch64 8.32-30.amzn2023.0.2         amazonlinux 2.0 M
 cpio                    aarch64 2.13-10.amzn2023.0.1         amazonlinux 269 k
 cracklib                aarch64 2.9.6-27.amzn2023.0.1        amazonlinux  83 k
```

```
 cracklib-dicts          aarch64 2.9.6-27.amzn2023.0.1        amazonlinux 3.6 M
 crontabs                noarch  1.11-24.20190603git.amzn2023.0.1
                                                              amazonlinux  19 k
 crypto-policies         noarch  20230128-1.gitdfb10ea.amzn2023.0.1
                                                              amazonlinux  61 k
 crypto-policies-scripts noarch  20230128-1.gitdfb10ea.amzn2023.0.1
                                                              amazonlinux  81 k
...
Installing dependencies:
 amazon-linux-repo-cdn   noarch  2023.0.20230210-0.amzn2023   amazonlinux  16 k
 xxhash-libs             aarch64 0.8.0-3.amzn2023.0.1         amazonlinux  32 k
Installing weak dependencies:
 amazon-chrony-config    noarch  4.2-7.amzn2023.0.4           amazonlinux  14 k
 gawk-all-langpacks      aarch64 5.1.0-3.amzn2023.0.1         amazonlinux 207 k

Transaction Summary
================================================================================
Install    5 Packages
Upgrade  413 Packages

Total download size: 199 M
```

Note

Se você usou a variável de substituição `/etc/dnf/vars/releasever`, use o comando a seguir para restaurar o comportamento de bloqueio padrão apagando o valor de substituição.

```
$ sudo rm /etc/dnf/vars/releasever
```

O uso de uma substituição persistente ao uso `latest` em vez de uma versão específica é semelhante ao comportamento padrão do. AL2 Existem serviços criados AMIs com base nos AL2 quais desabilitam esse comportamento e bloqueiam versões de pacotes específicas, como você obtém por padrão em AL2 023.

Em vez de desativar as atualizações determinísticas, recomendamos substituir as instâncias por aquelas iniciadas a partir de uma nova AMI. Se a substituição da instância não for uma opção, recomendamos o uso de ferramentas como o AWS Systems Manager Patch Manager para orquestrar a aplicação de atualizações em uma frota. EC2 O Image Builder também pode criar, corrigir e testar automaticamente suas próprias imagens AMIs derivadas de AL2 023 imagens

básicas. Você também pode usar Receba notificações sobre novas atualizações o que pode ser usado para acionar seus próprios pipelines de construção de AMI.

O uso `latest` em um ambiente de pré-produção e, em seguida, a implantação no uso de produção `latest` não oferece proteção contra nenhum problema entre uma atualização do sistema operacional e seu aplicativo. Uma nova versão AL2 023 pode ser lançada a qualquer momento e, portanto, todos os usos `latest` na produção acarretam riscos.

Gerencie atualizações de pacotes e sistemas operacionais em AL2 023

Diferentemente das versões anteriores do Amazon Linux, o AL2 023 AMIs está bloqueado em uma versão específica do repositório Amazon Linux. Para aplicar correções de segurança e de erros a uma instância AL2 023, atualize o DNF configuração para a versão de lançamento mais recente disponível. Como alternativa, execute uma instância AL2 023 mais recente.

Esta seção descreve como gerenciar DNF pacotes e repositórios em uma instância em execução. Ele também descreve como configurar DNF de um script de dados do usuário para habilitar o repositório Amazon Linux mais recente disponível no momento do lançamento. Para obter mais informações, consulte Referência de comandos da DNF.

É recomendável aplicar todas as atualizações disponíveis em uma nova versão AL2 023. Escolher apenas atualizações de segurança ou apenas atualizações específicas deve ser a exceção e não a regra. Para listar quais Consultorias de segurança são relevantes para uma instância específica, consulteListando os avisos aplicáveis. Para obter informações sobre como instalar somente atualizações relevantes para um comunicado específico, consulteAplicando atualizações de segurança no local.

> **Important**
>
> Se você quiser denunciar uma vulnerabilidade ou tiver uma preocupação de segurança em relação a serviços em AWS nuvem ou projetos de código aberto, entre em contato com a AWS Segurança usando a página Relatórios de vulnerabilidades

Tópicos

- Verificar as atualizações de pacotes disponíveis

- Aplicando atualizações de segurança usando DNF e versões do repositório
- Reinício automático do serviço após atualizações (de segurança)
- Quando é necessário reinicializar para aplicar as atualizações de segurança?
- Lançamento de uma instância com a versão mais recente do repositório ativada
- Obtendo informações de suporte do pacote
- Verificando as versões mais recentes do repositório com dnf check-release-update
- Adicionar, habilitar ou desabilitar novos repositórios
- Adicionando repositórios com cloud-init

Verificar as atualizações de pacotes disponíveis

Você pode usar o comando `dnf check-update` para verificar se há atualizações no seu sistema. Para AL2 023, recomendamos que você adicione a `--releasever=`*`version-number`* opção ao comando.

Quando você adiciona essa opção, DNF também verifica se há atualizações para uma versão posterior do repositório. Por exemplo, depois de executar o comando `dnf check-update`, use a última versão retornada como o valor para o *`version-number`*.

Se a instância for atualizada para usar a versão mais recente do repositório, a saída incluirá uma lista de todos os pacotes a serem atualizados.

> **Note**
>
> Se você não especificar a versão de lançamento com o sinalizador opcional no comando `dnf check-update`, somente a versão do repositório atualmente configurada será verificada. Isso significa que os pacotes na versão posterior do repositório não são verificados.

Updates in a specific version

Neste exemplo, veremos quais atualizações estão disponíveis na versão 2023.1.20230628 se lançarmos um contêiner da versão 2023.0.20230315.

> **Note**
>
> Este exemplo usa as versões 2023.0.20230315 e 2023.1.20230628, e essas não são a versão mais recente da 023. Consulte as notas de versão AL2 023 para ver as versões mais recentes, que contêm as AL2 atualizações de segurança mais recentes.

Neste exemplo, começaremos com uma imagem de contêiner para a versão 2023.0.20230315.

Primeiro, buscamos essa imagem do contêiner no registro do contêiner. O `.0` final indica a versão da imagem para uma versão específica; essa versão da imagem geralmente é zero.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
   2023.0.20230315.0: Pulling from amazonlinux/amazonlinux
b76f3b09316a: Pull complete
Digest: sha256:94e7183b0739140dbd5b639fb7600f0a2299cec5df8780c26d9cb409da5315a9
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.0.20230315.0
public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

Agora podemos gerar uma concha dentro do contêiner, a partir da qual verificaremos se há atualizações.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
   bash-5.2#
```

O `dnf check-update` comando agora é usado para verificar as atualizações disponíveis na versão 2023.1.20230628.

> **Note**
>
> A aplicação de atualizações de pacotes é uma operação privilegiada. Embora a elevação de privilégios normalmente não seja necessária ao executar em um contêiner, se estiver executando em um ambiente não conteinerizado, como uma EC2 instância da Amazon, você pode verificar se há atualizações sem elevar os privilégios.

```
$ dnf check-update --releasever=2023.1.20230628
```

```
Amazon Linux 2023 repository                    60 MB/s |  15 MB     00:00
Last metadata expiration check: 0:00:02 ago on Mon Jul 22 17:25:34 2024.

amazon-linux-repo-cdn.noarch        2023.1.20230628-0.amzn2023        amazonlinux
ca-certificates.noarch              2023.2.60-1.0.amzn2023.0.2        amazonlinux
curl-minimal.x86_64                 8.0.1-1.amzn2023                  amazonlinux
glib2.x86_64                        2.74.7-688.amzn2023.0.1           amazonlinux
glibc.x86_64                        2.34-52.amzn2023.0.3              amazonlinux
glibc-common.x86_64                 2.34-52.amzn2023.0.3              amazonlinux
glibc-minimal-langpack.x86_64       2.34-52.amzn2023.0.3              amazonlinux
gnupg2-minimal.x86_64               2.3.7-1.amzn2023.0.4              amazonlinux
keyutils-libs.x86_64                1.6.3-1.amzn2023                  amazonlinux
libcap.x86_64                       2.48-2.amzn2023.0.3               amazonlinux
libcurl-minimal.x86_64              8.0.1-1.amzn2023                  amazonlinux
libgcc.x86_64                       11.3.1-4.amzn2023.0.3             amazonlinux
libgomp.x86_64                      11.3.1-4.amzn2023.0.3             amazonlinux
libstdc++.x86_64                    11.3.1-4.amzn2023.0.3             amazonlinux
libxml2.x86_64                      2.10.4-1.amzn2023.0.1             amazonlinux
ncurses-base.noarch                 6.2-4.20200222.amzn2023.0.4       amazonlinux
ncurses-libs.x86_64                 6.2-4.20200222.amzn2023.0.4       amazonlinux
openssl-libs.x86_64                 1:3.0.8-1.amzn2023.0.3            amazonlinux
python3-rpm.x86_64                  4.16.1.3-12.amzn2023.0.6          amazonlinux
rpm.x86_64                          4.16.1.3-12.amzn2023.0.6          amazonlinux
rpm-build-libs.x86_64               4.16.1.3-12.amzn2023.0.6          amazonlinux
rpm-libs.x86_64                     4.16.1.3-12.amzn2023.0.6          amazonlinux
rpm-sign-libs.x86_64                4.16.1.3-12.amzn2023.0.6          amazonlinux
system-release.noarch               2023.1.20230628-0.amzn2023        amazonlinux
tzdata.noarch                       2023c-1.amzn2023.0.1              amazonlinux
bash-5.2#
```

A versão do `system-release` pacote mostra a versão para a qual um `dnf upgrade` comando seria atualizado, que é a versão 2023.1.20230628 solicitada no comando. `dnf check-update --releasever=`*`2023.1.20230628`*

Updates in the latest version

Neste exemplo, veremos quais atualizações estão disponíveis na versão AL2 023 se lançarmos um contêiner da `latest` versão 2023.4.20240319. No momento em que este artigo foi escrito, a `latest` versão era 2023.5.20240708, portanto, as atualizações listadas neste exemplo serão a partir dessa versão.

> **Note**
>
> Este exemplo usa as versões 2023.4.20240319 e 2023.5.20240708, sendo a última a versão mais recente no momento em que este artigo foi escrito. Para obter mais informações sobre as versões mais recentes, consulte as notas de versão AL2 023.

Neste exemplo, começaremos com uma imagem de contêiner para a versão 2023.4.20240319.

Primeiro, buscamos essa imagem do contêiner no registro do contêiner. O `.1` final indica a versão da imagem para uma versão específica. Embora a versão da imagem geralmente seja zero, este exemplo usa uma versão em que a versão da imagem é uma.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
   2023.4.20240319.1: Pulling from amazonlinux/amazonlinux
6de065fda9a2: Pull complete
Digest: sha256:b4838c4cc9211d966b6ea158dacc9eda7433a16ba94436508c2d9f01f7658b4e
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.4.20240319.1
public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

Agora podemos gerar uma concha dentro do contêiner, a partir da qual verificaremos se há atualizações.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
   bash-5.2#
```

O `dnf check-update` comando agora é usado para verificar as atualizações disponíveis na `latest` versão, que no momento da redação era 2023.5.20240708.

> **Note**
>
> A aplicação de atualizações de pacotes é uma operação privilegiada. Embora a elevação de privilégios normalmente não seja necessária ao executar em um contêiner, se estiver executando em um ambiente não conteinerizado, como uma EC2 instância da Amazon, você pode verificar se há atualizações sem elevar os privilégios.

```
$ dnf --releasever=latest check-update
```

```
Amazon Linux 2023 repository                    78 MB/s |  25 MB     00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 17:39:13 2024.

amazon-linux-repo-cdn.noarch        2023.5.20240708-1.amzn2023        amazonlinux
curl-minimal.x86_64                 8.5.0-1.amzn2023.0.4              amazonlinux
dnf.noarch                          4.14.0-1.amzn2023.0.5             amazonlinux
dnf-data.noarch                     4.14.0-1.amzn2023.0.5             amazonlinux
expat.x86_64                        2.5.0-1.amzn2023.0.4              amazonlinux
glibc.x86_64                        2.34-52.amzn2023.0.10             amazonlinux
glibc-common.x86_64                 2.34-52.amzn2023.0.10             amazonlinux
glibc-minimal-langpack.x86_64       2.34-52.amzn2023.0.10             amazonlinux
krb5-libs.x86_64                    1.21-3.amzn2023.0.4               amazonlinux
libblkid.x86_64                     2.37.4-1.amzn2023.0.4             amazonlinux
libcurl-minimal.x86_64              8.5.0-1.amzn2023.0.4              amazonlinux
libmount.x86_64                     2.37.4-1.amzn2023.0.4             amazonlinux
libnghttp2.x86_64                   1.59.0-3.amzn2023.0.1             amazonlinux
libsmartcols.x86_64                 2.37.4-1.amzn2023.0.4             amazonlinux
libuuid.x86_64                      2.37.4-1.amzn2023.0.4             amazonlinux
openssl-libs.x86_64                 1:3.0.8-1.amzn2023.0.12           amazonlinux
python3.x86_64                      3.9.16-1.amzn2023.0.8             amazonlinux
python3-dnf.noarch                  4.14.0-1.amzn2023.0.5             amazonlinux
python3-libs.x86_64                 3.9.16-1.amzn2023.0.8             amazonlinux
system-release.noarch               2023.5.20240708-1.amzn2023        amazonlinux
yum.noarch                          4.14.0-1.amzn2023.0.5             amazonlinux
bash-5.2#
```

A versão do `system-release` pacote mostra a versão para a qual um `dnf upgrade` comando seria atualizado.

Para esse comando, se houver pacotes mais novos disponíveis, o código de retorno será 100. Se não houver pacotes mais novos disponíveis, o código de retorno será 0. Além disso, a saída também lista todos os pacotes a serem atualizados.

Aplicando atualizações de segurança usando DNF e versões do repositório

Novas atualizações de pacotes e atualizações de segurança são disponibilizadas somente para novas versões do repositório. Para instâncias que você executou a partir de versões anteriores da AMI AL2 023, você deve atualizar a versão do repositório antes de poder instalar as atualizações de segurança. O comando `dnf check-release-update` inclui um exemplo de comando update que atualiza todos os pacotes instalados no sistema para versões em um repositório mais novo.

> **Note**
>
> Se você não especificar a versão de lançamento com o sinalizador opcional no comando `dnf check-update`, somente a versão do repositório atualmente configurada será verificada. Isso significa que qualquer atualização nos pacotes instalados presentes em qualquer versão posterior do repositório não é aplicada.

Esta seção aborda o caminho de atualização recomendado para aplicar todas as atualizações disponíveis em vez de escolher atualizações individuais ou somente aquelas marcadas como atualizações de segurança. Ao aplicar todas as atualizações, as instâncias existentes são movidas para o mesmo conjunto de pacotes da execução de uma AMI atualizada. Essa consistência reduz a variação das versões do pacote em uma frota. Para obter mais informações sobre como aplicar atualizações específicas, consulteAplicando atualizações de segurança no local.

Applying updates in a specific version

Neste exemplo, aplicaremos as atualizações disponíveis na versão 2023.1.20230628 se lançarmos um contêiner da versão 2023.0.20230315.

> **Note**
>
> Este exemplo usa as versões 2023.0.20230315 e 2023.1.20230628, e essas não são a versão mais recente da 023. Consulte as notas de versão AL2 023 para ver as versões mais recentes, que contêm as AL2 atualizações de segurança mais recentes.

Neste exemplo, começaremos com uma imagem de contêiner para a versão 2023.0.20230315.

Primeiro, buscamos essa imagem do contêiner no registro do contêiner. O `.0` final indica a versão da imagem para uma versão específica; essa versão da imagem geralmente é zero.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
   2023.0.20230315.0: Pulling from amazonlinux/amazonlinux
b76f3b09316a: Pull complete
Digest: sha256:94e7183b0739140dbd5b639fb7600f0a2299cec5df8780c26d9cb409da5315a9
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.0.20230315.0
public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

Agora podemos gerar uma concha dentro do contêiner, a partir da qual aplicaremos as atualizações.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
   bash-5.2#
```

O `dnf upgrade` comando agora é usado para aplicar todas as atualizações presentes na versão 2023.1.20230628.

> **Note**
>
> A aplicação de atualizações de pacotes é uma operação privilegiada. Embora a elevação de privilégios normalmente não seja necessária ao executar em um contêiner, se estiver executando em um ambiente não conteinerizado, como uma EC2 instância da Amazon, você precisará executar o `dnf upgrade` comando como usuário. `root` Isso pode ser feito usando os `su` comandos `sudo` ou.

```
$ dnf upgrade --releasever=2023.1.20230628
Amazon Linux 2023 repository                    38 MB/s |  15 MB     00:00
Last metadata expiration check: 0:00:02 ago on Mon Jul 22 17:49:08 2024.
Dependencies resolved.
================================================================================
 Package                 Arch    Version                       Repository    Size
================================================================================
Upgrading:
 amazon-linux-repo-cdn   noarch  2023.1.20230628-0.amzn2023    amazonlinux   18 k
 ca-certificates         noarch  2023.2.60-1.0.amzn2023.0.2    amazonlinux  829 k
 curl-minimal            x86_64  8.0.1-1.amzn2023              amazonlinux  150 k
 glib2                   x86_64  2.74.7-688.amzn2023.0.1       amazonlinux  2.7 M
 glibc                   x86_64  2.34-52.amzn2023.0.3          amazonlinux  1.9 M
 glibc-common            x86_64  2.34-52.amzn2023.0.3          amazonlinux  307 k
 glibc-minimal-langpack  x86_64  2.34-52.amzn2023.0.3          amazonlinux   35 k
 gnupg2-minimal          x86_64  2.3.7-1.amzn2023.0.4          amazonlinux  421 k
 keyutils-libs           x86_64  1.6.3-1.amzn2023              amazonlinux   33 k
 libcap                  x86_64  2.48-2.amzn2023.0.3           amazonlinux   67 k
 libcurl-minimal         x86_64  8.0.1-1.amzn2023              amazonlinux  249 k
 libgcc                  x86_64  11.3.1-4.amzn2023.0.3         amazonlinux  105 k
 libgomp                 x86_64  11.3.1-4.amzn2023.0.3         amazonlinux  280 k
 libstdc++               x86_64  11.3.1-4.amzn2023.0.3         amazonlinux  744 k
 libxml2                 x86_64  2.10.4-1.amzn2023.0.1         amazonlinux  706 k
```

```
 ncurses-base            noarch  6.2-4.20200222.amzn2023.0.4  amazonlinux   60 k
 ncurses-libs            x86_64  6.2-4.20200222.amzn2023.0.4  amazonlinux  328 k
 openssl-libs            x86_64  1:3.0.8-1.amzn2023.0.3       amazonlinux  2.2 M
 python3-rpm             x86_64  4.16.1.3-12.amzn2023.0.6     amazonlinux   88 k
 rpm                     x86_64  4.16.1.3-12.amzn2023.0.6     amazonlinux  486 k
 rpm-build-libs          x86_64  4.16.1.3-12.amzn2023.0.6     amazonlinux   90 k
 rpm-libs                x86_64  4.16.1.3-12.amzn2023.0.6     amazonlinux  309 k
 rpm-sign-libs           x86_64  4.16.1.3-12.amzn2023.0.6     amazonlinux   21 k
 system-release          noarch  2023.1.20230628-0.amzn2023   amazonlinux   29 k
 tzdata                  noarch  2023c-1.amzn2023.0.1         amazonlinux  433 k

Transaction Summary
================================================================================
Upgrade  25 Packages

Total download size: 12 M
Is this ok [y/N]:
```

A versão do `system-release` pacote mostra a versão para a qual um `dnf upgrade` comando seria atualizado, que é a versão 2023.1.20230628 solicitada no comando. `dnf upgrade --releasever=`*`2023.1.20230628`*

Por padrão, `dnf` solicitará que você confirme que deseja aplicar as atualizações. Você pode ignorar esse prompt usando o `-y` sinalizador para`dnf`. Neste exemplo, o `dnf upgrade -y --releasever=`*`2023.1.20230628`* comando não pediria confirmação antes de aplicar as atualizações. Isso é útil em scripts ou outros ambientes de automação.

Depois de confirmar que você deseja aplicar as atualizações, `dnf` aplique-as.

```
Is this ok [y/N]:y
   Downloading Packages:
(1/25): libcap-2.48-2.amzn2023.0.3.x86_64.rpm    1.5 MB/s |  67 kB     00:00
(2/25): python3-rpm-4.16.1.3-12.amzn2023.0.6.x86 2.1 MB/s |  88 kB     00:00
(3/25): libcurl-minimal-8.0.1-1.amzn2023.x86_64. 2.6 MB/s | 249 kB     00:00
(4/25): glib2-2.74.7-688.amzn2023.0.1.x86_64.rpm  26 MB/s | 2.7 MB     00:00
(5/25): glibc-minimal-langpack-2.34-52.amzn2023. 1.3 MB/s |  35 kB     00:00
(6/25): rpm-build-libs-4.16.1.3-12.amzn2023.0.6. 2.8 MB/s |  90 kB     00:00
(7/25): rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64 6.6 MB/s | 309 kB     00:00
(8/25): libgcc-11.3.1-4.amzn2023.0.3.x86_64.rpm  3.9 MB/s | 105 kB     00:00
(9/25): glibc-common-2.34-52.amzn2023.0.3.x86_64  11 MB/s | 307 kB     00:00
(10/25): glibc-2.34-52.amzn2023.0.3.x86_64.rpm    31 MB/s | 1.9 MB     00:00
(11/25): rpm-sign-libs-4.16.1.3-12.amzn2023.0.6. 877 kB/s |  21 kB     00:00
(12/25): gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86  15 MB/s | 421 kB     00:00
```

```
(13/25): openssl-libs-3.0.8-1.amzn2023.0.3.x86_6  35 MB/s | 2.2 MB     00:00
(14/25): libxml2-2.10.4-1.amzn2023.0.1.x86_64.rp  14 MB/s | 706 kB     00:00
(15/25): curl-minimal-8.0.1-1.amzn2023.x86_64.rp 4.2 MB/s | 150 kB     00:00
(16/25): rpm-4.16.1.3-12.amzn2023.0.6.x86_64.rpm  11 MB/s | 486 kB     00:00
(17/25): libgomp-11.3.1-4.amzn2023.0.3.x86_64.rp 7.0 MB/s | 280 kB     00:00
(18/25): libstdc++-11.3.1-4.amzn2023.0.3.x86_64.  14 MB/s | 744 kB     00:00
(19/25): keyutils-libs-1.6.3-1.amzn2023.x86_64.r 1.6 MB/s |  33 kB     00:00
(20/25): ncurses-libs-6.2-4.20200222.amzn2023.0.  10 MB/s | 328 kB     00:00
(21/25): tzdata-2023c-1.amzn2023.0.1.noarch.rpm   11 MB/s | 433 kB     00:00
(22/25): amazon-linux-repo-cdn-2023.1.20230628-0 781 kB/s |  18 kB     00:00
(23/25): ca-certificates-2023.2.60-1.0.amzn2023.  16 MB/s | 829 kB     00:00
(24/25): system-release-2023.1.20230628-0.amzn20 1.5 MB/s |  29 kB     00:00
(25/25): ncurses-base-6.2-4.20200222.amzn2023.0. 3.1 MB/s |  60 kB     00:00
--------------------------------------------------------------------------------
Total                                            28 MB/s |  12 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Upgrading        : libgcc-11.3.1-4.amzn2023.0.3.x86_64                   1/50
  Running scriptlet: libgcc-11.3.1-4.amzn2023.0.3.x86_64                   1/50
  Upgrading        : system-release-2023.1.20230628-0.amzn2023.noarch       2/50
  Upgrading        : amazon-linux-repo-cdn-2023.1.20230628-0.amzn2023.no    3/50
  Upgrading        : ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch        4/50
  Upgrading        : tzdata-2023c-1.amzn2023.0.1.noarch                     5/50
  Upgrading        : glibc-common-2.34-52.amzn2023.0.3.x86_64               6/50
  Running scriptlet: glibc-2.34-52.amzn2023.0.3.x86_64                      7/50
  Upgrading        : glibc-2.34-52.amzn2023.0.3.x86_64                      7/50
  Running scriptlet: glibc-2.34-52.amzn2023.0.3.x86_64                      7/50
  Upgrading        : glibc-minimal-langpack-2.34-52.amzn2023.0.3.x86_64     8/50
  Upgrading        : libcap-2.48-2.amzn2023.0.3.x86_64                      9/50
  Upgrading        : gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86_64            10/50
  Upgrading        : libgomp-11.3.1-4.amzn2023.0.3.x86_64                  11/50
  Running scriptlet: ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch     12/50
  Upgrading        : ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch     12/50
  Running scriptlet: ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch     12/50
  Upgrading        : openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64            13/50
  Upgrading        : libcurl-minimal-8.0.1-1.amzn2023.x86_64               14/50
  Upgrading        : curl-minimal-8.0.1-1.amzn2023.x86_64                  15/50
  Upgrading        : rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64              16/50
  Upgrading        : rpm-4.16.1.3-12.amzn2023.0.6.x86_64                   17/50
  Upgrading        : rpm-build-libs-4.16.1.3-12.amzn2023.0.6.x86_64        18/50
```

```
  Upgrading        : rpm-sign-libs-4.16.1.3-12.amzn2023.0.6.x86_64          19/50
  Upgrading        : python3-rpm-4.16.1.3-12.amzn2023.0.6.x86_64            20/50
  Upgrading        : glib2-2.74.7-688.amzn2023.0.1.x86_64                   21/50
  Upgrading        : libxml2-2.10.4-1.amzn2023.0.1.x86_64                   22/50
  Upgrading        : libstdc++-11.3.1-4.amzn2023.0.3.x86_64                 23/50
  Upgrading        : keyutils-libs-1.6.3-1.amzn2023.x86_64                  24/50
  Upgrading        : ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64        25/50
  Cleanup          : glib2-2.73.2-680.amzn2023.0.3.x86_64                   26/50
  Cleanup          : libstdc++-11.3.1-4.amzn2023.0.2.x86_64                 27/50
  Cleanup          : libxml2-2.10.3-2.amzn2023.0.1.x86_64                   28/50
  Cleanup          : python3-rpm-4.16.1.3-12.amzn2023.0.5.x86_64            29/50
  Cleanup          : rpm-build-libs-4.16.1.3-12.amzn2023.0.5.x86_64         30/50
  Cleanup          : rpm-sign-libs-4.16.1.3-12.amzn2023.0.5.x86_64          31/50
  Cleanup          : rpm-libs-4.16.1.3-12.amzn2023.0.5.x86_64               32/50
  Cleanup          : libcap-2.48-2.amzn2023.0.2.x86_64                      33/50
  Cleanup          : gnupg2-minimal-2.3.7-1.amzn2023.0.3.x86_64             34/50
  Cleanup          : ncurses-libs-6.2-4.20200222.amzn2023.0.3.x86_64        35/50
  Cleanup          : libgomp-11.3.1-4.amzn2023.0.2.x86_64                   36/50
  Cleanup          : rpm-4.16.1.3-12.amzn2023.0.5.x86_64                    37/50
  Cleanup          : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64              38/50
  Cleanup          : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64           39/50
  Cleanup          : openssl-libs-1:3.0.8-1.amzn2023.0.1.x86_64             40/50
  Cleanup          : keyutils-libs-1.6.1-2.amzn2023.0.2.x86_64              41/50
  Cleanup          : amazon-linux-repo-cdn-2023.0.20230315-1.amzn2023.no    42/50
  Cleanup          : system-release-2023.0.20230315-1.amzn2023.noarch       43/50
  Cleanup          : ca-certificates-2023.2.60-1.0.amzn2023.0.1.noarch      44/50
  Cleanup          : ncurses-base-6.2-4.20200222.amzn2023.0.3.noarch        45/50
  Cleanup          : glibc-minimal-langpack-2.34-52.amzn2023.0.2.x86_64     46/50
  Cleanup          : glibc-2.34-52.amzn2023.0.2.x86_64                      47/50
  Cleanup          : glibc-common-2.34-52.amzn2023.0.2.x86_64               48/50
  Cleanup          : tzdata-2022g-1.amzn2023.0.1.noarch                     49/50
  Cleanup          : libgcc-11.3.1-4.amzn2023.0.2.x86_64                    50/50
Running scriptlet: libgcc-11.3.1-4.amzn2023.0.2.x86_64                    50/50
Running scriptlet: ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch       50/50
Running scriptlet: rpm-4.16.1.3-12.amzn2023.0.6.x86_64                     50/50
Running scriptlet: libgcc-11.3.1-4.amzn2023.0.2.x86_64                     50/50
  Verifying        : libcurl-minimal-8.0.1-1.amzn2023.x86_64                 1/50
  Verifying        : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64            2/50
  Verifying        : libcap-2.48-2.amzn2023.0.3.x86_64                       3/50
  Verifying        : libcap-2.48-2.amzn2023.0.2.x86_64                       4/50
  Verifying        : glib2-2.74.7-688.amzn2023.0.1.x86_64                    5/50
  Verifying        : glib2-2.73.2-680.amzn2023.0.3.x86_64                    6/50
  Verifying        : python3-rpm-4.16.1.3-12.amzn2023.0.6.x86_64             7/50
  Verifying        : python3-rpm-4.16.1.3-12.amzn2023.0.5.x86_64             8/50
```

```
  Verifying        : glibc-minimal-langpack-2.34-52.amzn2023.0.3.x86_64     9/50
  Verifying        : glibc-minimal-langpack-2.34-52.amzn2023.0.2.x86_64    10/50
  Verifying        : rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64              11/50
  Verifying        : rpm-libs-4.16.1.3-12.amzn2023.0.5.x86_64              12/50
  Verifying        : rpm-build-libs-4.16.1.3-12.amzn2023.0.6.x86_64        13/50
  Verifying        : rpm-build-libs-4.16.1.3-12.amzn2023.0.5.x86_64        14/50
  Verifying        : glibc-2.34-52.amzn2023.0.3.x86_64                     15/50
  Verifying        : glibc-2.34-52.amzn2023.0.2.x86_64                     16/50
  Verifying        : libgcc-11.3.1-4.amzn2023.0.3.x86_64                   17/50
  Verifying        : libgcc-11.3.1-4.amzn2023.0.2.x86_64                   18/50
  Verifying        : glibc-common-2.34-52.amzn2023.0.3.x86_64              19/50
  Verifying        : glibc-common-2.34-52.amzn2023.0.2.x86_64              20/50
  Verifying        : rpm-sign-libs-4.16.1.3-12.amzn2023.0.6.x86_64         21/50
  Verifying        : rpm-sign-libs-4.16.1.3-12.amzn2023.0.5.x86_64         22/50
  Verifying        : openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64            23/50
  Verifying        : openssl-libs-1:3.0.8-1.amzn2023.0.1.x86_64            24/50
  Verifying        : gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86_64            25/50
  Verifying        : gnupg2-minimal-2.3.7-1.amzn2023.0.3.x86_64            26/50
  Verifying        : libxml2-2.10.4-1.amzn2023.0.1.x86_64                  27/50
  Verifying        : libxml2-2.10.3-2.amzn2023.0.1.x86_64                  28/50
  Verifying        : curl-minimal-8.0.1-1.amzn2023.x86_64                  29/50
  Verifying        : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64             30/50
  Verifying        : rpm-4.16.1.3-12.amzn2023.0.6.x86_64                   31/50
  Verifying        : rpm-4.16.1.3-12.amzn2023.0.5.x86_64                   32/50
  Verifying        : libstdc++-11.3.1-4.amzn2023.0.3.x86_64                33/50
  Verifying        : libstdc++-11.3.1-4.amzn2023.0.2.x86_64                34/50
  Verifying        : libgomp-11.3.1-4.amzn2023.0.3.x86_64                  35/50
  Verifying        : libgomp-11.3.1-4.amzn2023.0.2.x86_64                  36/50
  Verifying        : keyutils-libs-1.6.3-1.amzn2023.x86_64                 37/50
  Verifying        : keyutils-libs-1.6.1-2.amzn2023.0.2.x86_64             38/50
  Verifying        : ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64       39/50
  Verifying        : ncurses-libs-6.2-4.20200222.amzn2023.0.3.x86_64       40/50
  Verifying        : ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch     41/50
  Verifying        : ca-certificates-2023.2.60-1.0.amzn2023.0.1.noarch     42/50
  Verifying        : tzdata-2023c-1.amzn2023.0.1.noarch                    43/50
  Verifying        : tzdata-2022g-1.amzn2023.0.1.noarch                    44/50
  Verifying        : amazon-linux-repo-cdn-2023.1.20230628-0.amzn2023.no   45/50
  Verifying        : amazon-linux-repo-cdn-2023.0.20230315-1.amzn2023.no   46/50
  Verifying        : system-release-2023.1.20230628-0.amzn2023.noarch       47/50
  Verifying        : system-release-2023.0.20230315-1.amzn2023.noarch       48/50
  Verifying        : ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch       49/50
  Verifying        : ncurses-base-6.2-4.20200222.amzn2023.0.3.noarch       50/50

Upgraded:
```

```
  amazon-linux-repo-cdn-2023.1.20230628-0.amzn2023.noarch
  ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch
  curl-minimal-8.0.1-1.amzn2023.x86_64
  glib2-2.74.7-688.amzn2023.0.1.x86_64
  glibc-2.34-52.amzn2023.0.3.x86_64
  glibc-common-2.34-52.amzn2023.0.3.x86_64
  glibc-minimal-langpack-2.34-52.amzn2023.0.3.x86_64
  gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86_64
  keyutils-libs-1.6.3-1.amzn2023.x86_64
  libcap-2.48-2.amzn2023.0.3.x86_64
  libcurl-minimal-8.0.1-1.amzn2023.x86_64
  libgcc-11.3.1-4.amzn2023.0.3.x86_64
  libgomp-11.3.1-4.amzn2023.0.3.x86_64
  libstdc++-11.3.1-4.amzn2023.0.3.x86_64
  libxml2-2.10.4-1.amzn2023.0.1.x86_64
  ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch
  ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64
  openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64
  python3-rpm-4.16.1.3-12.amzn2023.0.6.x86_64
  rpm-4.16.1.3-12.amzn2023.0.6.x86_64
  rpm-build-libs-4.16.1.3-12.amzn2023.0.6.x86_64
  rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64
  rpm-sign-libs-4.16.1.3-12.amzn2023.0.6.x86_64
  system-release-2023.1.20230628-0.amzn2023.noarch
  tzdata-2023c-1.amzn2023.0.1.noarch

  Complete!
bash-5.2#
```

Updates in the latest version

Neste exemplo, aplicaremos as atualizações disponíveis na `latest` versão AL2 023 se lançarmos um contêiner da versão 2023.4.20240319. No momento em que este artigo foi escrito, a `latest` versão era 2023.5.20240708, portanto, as atualizações listadas neste exemplo serão a partir dessa versão.

> **Note**
>
> Este exemplo usa as versões 2023.4.20240319 e 2023.5.20240708, sendo a última a versão mais recente no momento em que este artigo foi escrito. Para obter mais informações sobre as versões mais recentes, consulte as notas de versão AL2 023.

Neste exemplo, começaremos com uma imagem de contêiner para a versão 2023.4.20240319.

Primeiro, buscamos essa imagem do contêiner no registro do contêiner. O `.1` final indica a versão da imagem para uma versão específica. Embora a versão da imagem geralmente seja zero, este exemplo usa uma versão em que a versão da imagem é uma.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
   2023.4.20240319.1: Pulling from amazonlinux/amazonlinux
6de065fda9a2: Pull complete
Digest: sha256:b4838c4cc9211d966b6ea158dacc9eda7433a16ba94436508c2d9f01f7658b4e
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.4.20240319.1
public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

Agora podemos gerar uma concha dentro do contêiner, a partir da qual aplicaremos as atualizações.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
   bash-5.2#
```

O `dnf upgrade` comando agora é usado para aplicar as atualizações disponíveis na `latest` versão, que no momento da redação era 2023.5.20240708.

> **Note**
>
> A aplicação de atualizações de pacotes é uma operação privilegiada. Embora a elevação de privilégios normalmente não seja necessária ao executar em um contêiner, se estiver executando em um ambiente não conteinerizado, como uma EC2 instância da Amazon, você precisará executar o `dnf upgrade` comando como usuário. `root` Isso pode ser feito usando os `su` comandos `sudo` ou.

Por padrão, `dnf` solicitará que você confirme que deseja aplicar as atualizações. Neste exemplo, estamos ignorando esse prompt usando o `-y` sinalizador to`dnf`.

```
$ dnf -y --releasever=latest update
Amazon Linux 2023 repository                    75 MB/s |  25 MB     00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 18:00:10 2024.
Dependencies resolved.
================================================================================
```

```
 Package                 Arch    Version                        Repository    Size
================================================================================
Upgrading:
 amazon-linux-repo-cdn   noarch  2023.5.20240708-1.amzn2023   amazonlinux   17 k
 curl-minimal            x86_64  8.5.0-1.amzn2023.0.4         amazonlinux  160 k
 dnf                     noarch  4.14.0-1.amzn2023.0.5        amazonlinux  460 k
 dnf-data                noarch  4.14.0-1.amzn2023.0.5        amazonlinux   34 k
 expat                   x86_64  2.5.0-1.amzn2023.0.4         amazonlinux  117 k
 glibc                   x86_64  2.34-52.amzn2023.0.10        amazonlinux  1.9 M
 glibc-common            x86_64  2.34-52.amzn2023.0.10        amazonlinux  295 k
 glibc-minimal-langpack  x86_64  2.34-52.amzn2023.0.10        amazonlinux   23 k
 krb5-libs               x86_64  1.21-3.amzn2023.0.4          amazonlinux  758 k
 libblkid                x86_64  2.37.4-1.amzn2023.0.4        amazonlinux  105 k
 libcurl-minimal         x86_64  8.5.0-1.amzn2023.0.4         amazonlinux  275 k
 libmount                x86_64  2.37.4-1.amzn2023.0.4        amazonlinux  132 k
 libnghttp2              x86_64  1.59.0-3.amzn2023.0.1        amazonlinux   79 k
 libsmartcols            x86_64  2.37.4-1.amzn2023.0.4        amazonlinux   62 k
 libuuid                 x86_64  2.37.4-1.amzn2023.0.4        amazonlinux   26 k
 openssl-libs            x86_64  1:3.0.8-1.amzn2023.0.12      amazonlinux  2.2 M
 python3                 x86_64  3.9.16-1.amzn2023.0.8        amazonlinux   27 k
 python3-dnf             noarch  4.14.0-1.amzn2023.0.5        amazonlinux  409 k
 python3-libs            x86_64  3.9.16-1.amzn2023.0.8        amazonlinux  7.3 M
 system-release          noarch  2023.5.20240708-1.amzn2023   amazonlinux   28 k
 yum                     noarch  4.14.0-1.amzn2023.0.5        amazonlinux   32 k

 Transaction Summary
================================================================================
Upgrade  21 Packages

Total download size: 14 M
Downloading Packages:
(1/21): amazon-linux-repo-cdn-2023.5.20240708-1. 345 kB/s |  17 kB     00:00
(2/21): dnf-4.14.0-1.amzn2023.0.5.noarch.rpm     6.8 MB/s | 460 kB     00:00
(3/21): dnf-data-4.14.0-1.amzn2023.0.5.noarch.rp 1.6 MB/s |  34 kB     00:00
(4/21): expat-2.5.0-1.amzn2023.0.4.x86_64.rpm    4.6 MB/s | 117 kB     00:00
(5/21): glibc-2.34-52.amzn2023.0.10.x86_64.rpm    38 MB/s | 1.9 MB     00:00
(6/21): glibc-common-2.34-52.amzn2023.0.10.x86_6 8.8 MB/s | 295 kB     00:00
(7/21): glibc-minimal-langpack-2.34-52.amzn2023. 1.7 MB/s |  23 kB     00:00
(8/21): curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 998 kB/s | 160 kB     00:00
(9/21): libblkid-2.37.4-1.amzn2023.0.4.x86_64.rp 4.1 MB/s | 105 kB     00:00
(10/21): krb5-libs-1.21-3.amzn2023.0.4.x86_64.rp  16 MB/s | 758 kB     00:00
(11/21): libmount-2.37.4-1.amzn2023.0.4.x86_64.r 7.9 MB/s | 132 kB     00:00
(12/21): libnghttp2-1.59.0-3.amzn2023.0.1.x86_64 5.6 MB/s |  79 kB     00:00
(13/21): libsmartcols-2.37.4-1.amzn2023.0.4.x86_ 4.4 MB/s |  62 kB     00:00
```

```
(14/21): libcurl-minimal-8.5.0-1.amzn2023.0.4.x8 7.1 MB/s | 275 kB     00:00
(15/21): libuuid-2.37.4-1.amzn2023.0.4.x86_64.rp 1.1 MB/s |  26 kB     00:00
(16/21): python3-3.9.16-1.amzn2023.0.8.x86_64.rp 1.5 MB/s |  27 kB     00:00
(17/21): python3-dnf-4.14.0-1.amzn2023.0.5.noarc  19 MB/s | 409 kB     00:00
(18/21): system-release-2023.5.20240708-1.amzn20 1.9 MB/s |  28 kB     00:00
(19/21): yum-4.14.0-1.amzn2023.0.5.noarch.rpm    1.6 MB/s |  32 kB     00:00
(20/21): openssl-libs-3.0.8-1.amzn2023.0.12.x86_  26 MB/s | 2.2 MB     00:00
(21/21): python3-libs-3.9.16-1.amzn2023.0.8.x86_  59 MB/s | 7.3 MB     00:00
--------------------------------------------------------------------------------
Total                                             34 MB/s |  14 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Upgrading        : glibc-common-2.34-52.amzn2023.0.10.x86_64              1/42
  Upgrading        : glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64    2/42
  Running scriptlet: glibc-2.34-52.amzn2023.0.10.x86_64                     3/42
  Upgrading        : glibc-2.34-52.amzn2023.0.10.x86_64                     3/42
  Running scriptlet: glibc-2.34-52.amzn2023.0.10.x86_64                     3/42
  Upgrading        : libuuid-2.37.4-1.amzn2023.0.4.x86_64                   4/42
  Upgrading        : openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64            5/42
  Upgrading        : krb5-libs-1.21-3.amzn2023.0.4.x86_64                   6/42
  Upgrading        : libblkid-2.37.4-1.amzn2023.0.4.x86_64                  7/42
  Running scriptlet: libblkid-2.37.4-1.amzn2023.0.4.x86_64                  7/42
  Upgrading        : expat-2.5.0-1.amzn2023.0.4.x86_64                      8/42
  Upgrading        : python3-3.9.16-1.amzn2023.0.8.x86_64                   9/42
  Upgrading        : python3-libs-3.9.16-1.amzn2023.0.8.x86_64             10/42
  Upgrading        : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64               11/42
  Upgrading        : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64           12/42
  Upgrading        : system-release-2023.5.20240708-1.amzn2023.noarch      13/42
  Upgrading        : amazon-linux-repo-cdn-2023.5.20240708-1.amzn2023.no   14/42
  Upgrading        : dnf-data-4.14.0-1.amzn2023.0.5.noarch                 15/42
  Upgrading        : python3-dnf-4.14.0-1.amzn2023.0.5.noarch              16/42
  Upgrading        : dnf-4.14.0-1.amzn2023.0.5.noarch                      17/42
  Running scriptlet: dnf-4.14.0-1.amzn2023.0.5.noarch                      17/42
  Upgrading        : yum-4.14.0-1.amzn2023.0.5.noarch                      18/42
  Upgrading        : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64              19/42
  Upgrading        : libmount-2.37.4-1.amzn2023.0.4.x86_64                 20/42
  Upgrading        : libsmartcols-2.37.4-1.amzn2023.0.4.x86_64             21/42
  Cleanup          : yum-4.14.0-1.amzn2023.0.4.noarch                      22/42
  Running scriptlet: dnf-4.14.0-1.amzn2023.0.4.noarch                      23/42
  Cleanup          : dnf-4.14.0-1.amzn2023.0.4.noarch                      23/42
```

```
Running scriptlet: dnf-4.14.0-1.amzn2023.0.4.noarch                       23/42
Cleanup          : python3-dnf-4.14.0-1.amzn2023.0.4.noarch               24/42
Cleanup          : amazon-linux-repo-cdn-2023.4.20240319-1.amzn2023.no    25/42
Cleanup          : libmount-2.37.4-1.amzn2023.0.3.x86_64                  26/42
Cleanup          : curl-minimal-8.5.0-1.amzn2023.0.2.x86_64               27/42
Cleanup          : libcurl-minimal-8.5.0-1.amzn2023.0.2.x86_64            28/42
Cleanup          : krb5-libs-1.21-3.amzn2023.0.3.x86_64                   29/42
Cleanup          : libblkid-2.37.4-1.amzn2023.0.3.x86_64                  30/42
Cleanup          : libnghttp2-1.57.0-1.amzn2023.0.1.x86_64                31/42
Cleanup          : libsmartcols-2.37.4-1.amzn2023.0.3.x86_64              32/42
Cleanup          : system-release-2023.4.20240319-1.amzn2023.noarch       33/42
Cleanup          : dnf-data-4.14.0-1.amzn2023.0.4.noarch                  34/42
Cleanup          : python3-3.9.16-1.amzn2023.0.6.x86_64                   35/42
Cleanup          : python3-libs-3.9.16-1.amzn2023.0.6.x86_64              36/42
Cleanup          : openssl-libs-1:3.0.8-1.amzn2023.0.11.x86_64            37/42
Cleanup          : libuuid-2.37.4-1.amzn2023.0.3.x86_64                   38/42
Cleanup          : expat-2.5.0-1.amzn2023.0.3.x86_64                      39/42
Cleanup          : glibc-2.34-52.amzn2023.0.8.x86_64                      40/42
Cleanup          : glibc-minimal-langpack-2.34-52.amzn2023.0.8.x86_64     41/42
Cleanup          : glibc-common-2.34-52.amzn2023.0.8.x86_64               42/42
Running scriptlet: glibc-common-2.34-52.amzn2023.0.8.x86_64               42/42
Verifying        : amazon-linux-repo-cdn-2023.5.20240708-1.amzn2023.no     1/42
Verifying        : amazon-linux-repo-cdn-2023.4.20240319-1.amzn2023.no     2/42
Verifying        : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64                3/42
Verifying        : curl-minimal-8.5.0-1.amzn2023.0.2.x86_64                4/42
Verifying        : dnf-4.14.0-1.amzn2023.0.5.noarch                        5/42
Verifying        : dnf-4.14.0-1.amzn2023.0.4.noarch                        6/42
Verifying        : dnf-data-4.14.0-1.amzn2023.0.5.noarch                   7/42
Verifying        : dnf-data-4.14.0-1.amzn2023.0.4.noarch                   8/42
Verifying        : expat-2.5.0-1.amzn2023.0.4.x86_64                       9/42
Verifying        : expat-2.5.0-1.amzn2023.0.3.x86_64                      10/42
Verifying        : glibc-2.34-52.amzn2023.0.10.x86_64                     11/42
Verifying        : glibc-2.34-52.amzn2023.0.8.x86_64                      12/42
Verifying        : glibc-common-2.34-52.amzn2023.0.10.x86_64              13/42
Verifying        : glibc-common-2.34-52.amzn2023.0.8.x86_64               14/42
Verifying        : glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64    15/42
Verifying        : glibc-minimal-langpack-2.34-52.amzn2023.0.8.x86_64     16/42
Verifying        : krb5-libs-1.21-3.amzn2023.0.4.x86_64                   17/42
Verifying        : krb5-libs-1.21-3.amzn2023.0.3.x86_64                   18/42
Verifying        : libblkid-2.37.4-1.amzn2023.0.4.x86_64                  19/42
Verifying        : libblkid-2.37.4-1.amzn2023.0.3.x86_64                  20/42
Verifying        : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64            21/42
Verifying        : libcurl-minimal-8.5.0-1.amzn2023.0.2.x86_64            22/42
Verifying        : libmount-2.37.4-1.amzn2023.0.4.x86_64                  23/42
```

```
  Verifying        : libmount-2.37.4-1.amzn2023.0.3.x86_64                 24/42
  Verifying        : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64               25/42
  Verifying        : libnghttp2-1.57.0-1.amzn2023.0.1.x86_64               26/42
  Verifying        : libsmartcols-2.37.4-1.amzn2023.0.4.x86_64             27/42
  Verifying        : libsmartcols-2.37.4-1.amzn2023.0.3.x86_64             28/42
  Verifying        : libuuid-2.37.4-1.amzn2023.0.4.x86_64                  29/42
  Verifying        : libuuid-2.37.4-1.amzn2023.0.3.x86_64                  30/42
  Verifying        : openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64           31/42
  Verifying        : openssl-libs-1:3.0.8-1.amzn2023.0.11.x86_64           32/42
  Verifying        : python3-3.9.16-1.amzn2023.0.8.x86_64                  33/42
  Verifying        : python3-3.9.16-1.amzn2023.0.6.x86_64                  34/42
  Verifying        : python3-dnf-4.14.0-1.amzn2023.0.5.noarch              35/42
  Verifying        : python3-dnf-4.14.0-1.amzn2023.0.4.noarch              36/42
  Verifying        : python3-libs-3.9.16-1.amzn2023.0.8.x86_64             37/42
  Verifying        : python3-libs-3.9.16-1.amzn2023.0.6.x86_64             38/42
  Verifying        : system-release-2023.5.20240708-1.amzn2023.noarch      39/42
  Verifying        : system-release-2023.4.20240319-1.amzn2023.noarch      40/42
  Verifying        : yum-4.14.0-1.amzn2023.0.5.noarch                      41/42
  Verifying        : yum-4.14.0-1.amzn2023.0.4.noarch                      42/42

Upgraded:
  amazon-linux-repo-cdn-2023.5.20240708-1.amzn2023.noarch
  curl-minimal-8.5.0-1.amzn2023.0.4.x86_64
  dnf-4.14.0-1.amzn2023.0.5.noarch
  dnf-data-4.14.0-1.amzn2023.0.5.noarch
  expat-2.5.0-1.amzn2023.0.4.x86_64
  glibc-2.34-52.amzn2023.0.10.x86_64
  glibc-common-2.34-52.amzn2023.0.10.x86_64
  glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64
  krb5-libs-1.21-3.amzn2023.0.4.x86_64
  libblkid-2.37.4-1.amzn2023.0.4.x86_64
  libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64
  libmount-2.37.4-1.amzn2023.0.4.x86_64
  libnghttp2-1.59.0-3.amzn2023.0.1.x86_64
  libsmartcols-2.37.4-1.amzn2023.0.4.x86_64
  libuuid-2.37.4-1.amzn2023.0.4.x86_64
  openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64
  python3-3.9.16-1.amzn2023.0.8.x86_64
  python3-dnf-4.14.0-1.amzn2023.0.5.noarch
  python3-libs-3.9.16-1.amzn2023.0.8.x86_64
  system-release-2023.5.20240708-1.amzn2023.noarch
  yum-4.14.0-1.amzn2023.0.5.noarch

Complete!
```

```
bash-5.2#
```

Para descobrir as atualizações do AL2 023, faça um ou mais dos seguintes:

- Execute o comando `dnf check-update`. Isso verifica se há atualizações não aplicadas na versão do Amazon Linux à qual você está bloqueado. Isso pode mostrar atualizações se você atualizou somente o `system-release` pacote, movendo para qual versão dos repositórios a instância está bloqueada, mas sem aplicar nenhuma das atualizações disponíveis nela.
- Inscreva-se no tópico SNS de atualização do repositório Amazon Linux (`arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates`). Para obter instruções, consulte Assinatura de um tópico do Amazon SNS no Guia do desenvolvedor do Amazon Simple Notification Service.
- Consulte regularmente as notas de lançamento do AL2 023.
- Descubra novas versões porVerificando as versões mais recentes do repositório com `dnf check-release-update`.

> **Important**
>
> Novas versões do AL2 023 contendo atualizações de segurança são lançadas com frequência. Certifique-se de manter-se atualizado com os patches de segurança relevantes.

Reinício automático do serviço após atualizações (de segurança)

O Amazon Linux agora vem com o pacote smart-restart. `Smart-restart`reinicia os serviços systemd nas atualizações do sistema sempre que um pacote é instalado ou excluído usando o gerenciador de pacotes do sistema. Isso ocorre sempre que `dnf (update|upgrade|downgrade)` é executado.

`Smart-restart`usa o `needs-restarting` pacote de `dnf-utils` e um mecanismo personalizado de negação de listagem para determinar quais serviços precisam ser reiniciados e se a reinicialização do sistema é recomendada. Se uma reinicialização do sistema for recomendada, um arquivo marcador de dica de reinicialização será gerado (). `/run/smart-restart/reboot-hint-marker`

Para instalar o **`smart-restart`**

Execute o seguinte DNF comando (como você faria com qualquer outro pacote).

```
$ sudo dnf install smart-restart
```

Após a instalação, as transações subsequentes acionarão a `smart-restart` lógica.

Negar lista

`Smart-restart`pode ser instruído a impedir que determinados serviços sejam reiniciados. Os serviços bloqueados não contribuirão para a decisão de se uma reinicialização é necessária. Para bloquear serviços adicionais, adicione um arquivo com o sufixo `-denylist` in, `/etc/smart-restart-conf.d/` conforme mostrado no exemplo a seguir.

```
$ cat /etc/smart-restart-conf.d/custom-denylist
# Some comments
myservice.service
```

> **Note**
>
> Todos os `*-denylist` arquivos são lidos e avaliados ao decidir se uma reinicialização é necessária.

Ganchos personalizados

Além de negar a listagem, `smart-restart` fornece um mecanismo para executar scripts personalizados antes e depois das tentativas de reiniciar o serviço. Os scripts personalizados podem ser usados para executar manualmente as etapas de preparação ou para informar outros componentes sobre uma reinicialização restante ou concluída.

Todos os scripts `/etc/smart-restart-conf.d/` com o sufixo `-pre-restart` ou `-post-restart` são executados. Se a ordem for importante, prefixe todos os scripts com um número para garantir a ordem de execução, conforme mostrado no exemplo a seguir.

```
$ ls /etc/smart-restart-conf.d/*-pre-restart
001-my-script-pre-restart
002-some-other-script-pre-restart
```

Quando é necessário reinicializar para aplicar as atualizações de segurança?

Em algumas situações, o Amazon Linux exige uma reinicialização para aplicar as atualizações:

- As atualizações do pacote do kernel Linux exigem uma reinicialização para ativar o novo kernel com as atualizações de segurança mais recentes. O livepatching do Kernel pode permitir que você adie as atualizações de segurança por um período limitado de tempo. Para obter detalhes, consulteAtualização do Kernel Live em 023 AL2.
- Em instâncias EC2 Metal, o Amazon Linux fornece atualizações de microcódigo (por meio do `microcode_ctl` pacote para Intel CPUs e do `amd-ucode-firmware` pacote para AMD) CPUs. Essas atualizações de microcódigo só serão ativadas nas reinicializações subsequentes da instância. Para EC2 instâncias virtualizadas, o sistema AWS Nitro subjacente gerencia atualizações de microcódigo para você.
- Alguns serviços do systemd em execução só funcionarão corretamente após a reinicialização completa do sistema. O `smart-restart` mecanismo informará você sobre essas situações deixando dicas de reinicialização. Consulte Reinício automático do serviço após atualizações (de segurança).

Lançamento de uma instância com a versão mais recente do repositório ativada

Você pode adicionar DNF comandos para um script de dados do usuário para controlar o que RPM os pacotes são instalados em uma Amazon Linux AMI quando ela é lançada. No exemplo a seguir, um script de dados de usuário é usado para garantir que qualquer instância iniciada com o script de dados de usuário tenha as mesmas atualizações de pacote instaladas.

```
#!/bin/bash
dnf upgrade --releasever=2023.0.20230210
# Additional setup and install commands below
dnf install httpd php7.4 mysql80
```

Você deve executar esse script como superusuário (raiz). Para fazer isso, execute o comando a seguir.

```
$ sudo sh -c "bash nameofscript.sh"
```

Para obter mais informações, consulte Dados do usuário e scripts de shell no Guia EC2 do usuário da Amazon.

> **Note**
>
> Em vez de usar um script de dados do usuário, inicie a Amazon Linux AMI mais recente ou uma AMI personalizada baseada na Amazon Linux AMI. O Amazon Linux AMI mais recente tem todas as atualizações necessárias instaladas e está configurado para apontar para uma versão específica do repositório.

Obtendo informações de suporte do pacote

AL2O 023 incorpora muitos projetos diferentes de software de código aberto. Cada um desses projetos é gerenciado de forma independente do Amazon Linux e tem end-of-support lançamentos e cronogramas diferentes. Para fornecer a você informações específicas do Amazon Linux sobre esses diferentes pacotes, o DNF `supportinfo`O plugin fornece metadados sobre um pacote. No exemplo a seguir, o comando **`dnf supportinfo`** retorna metadados para o pacote `glibc`.

```
$ sudo dnf supportinfo --pkg glibc
Last metadata expiration check: 0:07:56 ago on Wed Mar  1 23:21:49 2023.
Name                 : glibc
Version              : 2.34-52.amzn2023.0.2
State                : installed
Support Status       : supported
Support Periods      : from 2023-03-15      : supported
                     : from 2028-03-15      : unsupported
Support Statement    : Amazon Linux 2023 End Of Life
Link                 : https://aws.amazon.com/amazon-linux-ami/faqs/
Other Info           : This is the support statement for AL2023. The
     ...: end of life of Amazon Linux 2023 would be March 2028.
     ...: From this point, the Amazon Linux 2023 packages (listed
  ...: below) will no longer, receive any updates from AWS.
```

As informações de suporte do Package também estão disponíveis na seção de declarações de suporte das notas de lançamento do AL2 023.

Verificando as versões mais recentes do repositório com **`dnf check-release-update`**

Em uma instância AL2 023, você pode usar o DNF utilitário para gerenciar repositórios e aplicar atualizações RPM pacotes. Esses pacotes estão disponíveis nos repositórios do Amazon Linux. Você pode usar o DNF comando `dnf check-release-update` para verificar se há novas versões do DNF repositório.

> **Note**
>
> AL2As imagens de contêiner 023 não incluem o `dnf check-release-update` comando por padrão.
>
> ```
> $ dnf check-release-update
> No such command: check-release-update. Please use /usr/bin/dnf --help
> It could be a DNF plugin command, try: "dnf install 'dnf-command(check-release-
> update)'"
> ```
>
> Quando `dnf install 'dnf-command(check-release-update)'` for executado, `dnf` instalará o pacote que fornece o `check-release-update` comando, que é o `dnf-plugin-release-notification` pacote. No exemplo abaixo, o `-q` argumento é dado `dnf` para que ele tenha uma saída silenciosa.
>
> ```
> $ dnf -y -q install 'dnf-command(check-release-update)'
> Installed:
>   dnf-plugin-release-notification-1.2-1.amzn2023.0.2.noarch
> ```

Em ambientes sem contêineres, como uma EC2 instância da Amazon, o `check-release-update` comando é incluído por padrão.

```
$ sudo dnf check-release-update
WARNING:
  A newer release of "Amazon Linux" is available.

  Available Versions:

  Version 2023.0.20230210:
    Run the following command to update to 2023.0.20230210:
```

```
      dnf upgrade --releasever=2023.0.20230210

    Release notes:
     https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes.html
```

Isso retorna uma lista completa de todas as versões mais recentes do DNF repositórios que estão disponíveis. Se nada for devolvido, isso significa que DNF está atualmente configurado para usar a versão mais recente disponível. A versão do `system-release` pacote atualmente instalado define o `releasever` DNF variável. Para verificar a versão atual do repositório, execute o comando a seguir.

```
$ rpm -q system-release --qf "%{VERSION}\n"
```

Quando você corre DNF Em transações de pacotes (como comandos de instalação, atualização ou remoção), uma mensagem de aviso notifica você sobre qualquer nova versão do repositório. Por exemplo, se você instalar o `httpd` pacote em uma instância que foi executada a partir de uma versão mais antiga da AL2 023, a saída a seguir será retornada.

```
$ sudo dnf install httpd -y
Last metadata expiration check: 0:16:52 ago on Wed Mar  1 23:21:49 2023.
Dependencies resolved.
================================================================================
 Package            Arch   Version                Repository   Size
================================================================================
Installing:
 httpd              x86_64 2.4.54-3.amzn2023.0.4  amazonlinux  46 k
Installing dependencies:
 apr                x86_64 1.7.2-2.amzn2023.0.2   amazonlinux 129 k
 apr-util           x86_64 1.6.3-1.amzn2023.0.1   amazonlinux  98 k
 generic-logos-httpd
                    noarch 18.0.0-12.amzn2023.0.3 amazonlinux  19 k
 httpd-core         x86_64 2.4.54-3.amzn2023.0.4  amazonlinux 1.3 M
 httpd-filesystem   noarch 2.4.54-3.amzn2023.0.4  amazonlinux  13 k
 httpd-tools        x86_64 2.4.54-3.amzn2023.0.4  amazonlinux  80 k
 libbrotli          x86_64 1.0.9-4.amzn2023.0.2   amazonlinux 315 k
 mailcap            noarch 2.1.49-3.amzn2023.0.3  amazonlinux  33 k
Installing weak dependencies:
 apr-util-openssl   x86_64 1.6.3-1.amzn2023.0.1   amazonlinux  17 k
 mod_http2          x86_64 1.15.24-1.amzn2023.0.3 amazonlinux 152 k
 mod_lua            x86_64 2.4.54-3.amzn2023.0.4  amazonlinux  60 k

Transaction Summary
```

```
================================================================================
Install  12 Packages

Total download size: 2.3 M
Installed size: 6.8 M
Downloading Packages:
(1/12): apr-util-openssl-1.6.3-1.am 212 kB/s |  17 kB     00:00
(2/12): apr-1.7.2-2.amzn2023.0.2.x8 1.1 MB/s | 129 kB     00:00
(3/12): httpd-core-2.4.54-3.amzn202 8.9 MB/s | 1.3 MB     00:00
(4/12): mod_http2-1.15.24-1.amzn202 1.9 MB/s | 152 kB     00:00
(5/12): apr-util-1.6.3-1.amzn2023.0 1.7 MB/s |  98 kB     00:00
(6/12): mod_lua-2.4.54-3.amzn2023.0 1.4 MB/s |  60 kB     00:00
(7/12): httpd-2.4.54-3.amzn2023.0.4 1.5 MB/s |  46 kB     00:00
(8/12): libbrotli-1.0.9-4.amzn2023. 4.4 MB/s | 315 kB     00:00
(9/12): mailcap-2.1.49-3.amzn2023.0 753 kB/s |  33 kB     00:00
(10/12): httpd-tools-2.4.54-3.amzn2 978 kB/s |  80 kB     00:00
(11/12): httpd-filesystem-2.4.54-3. 210 kB/s |  13 kB     00:00
(12/12): generic-logos-httpd-18.0.0 439 kB/s |  19 kB     00:00
--------------------------------------------------------------------------------
Total                                6.6 MB/s | 2.3 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Installing       : apr-1.7.2-2.amzn2023.0.2.x86_64                        1/12
  Installing       : apr-util-openssl-1.6.3-1.amzn2023.0.1.                 2/12
  Installing       : apr-util-1.6.3-1.amzn2023.0.1.x86_64                   3/12
  Installing       : mailcap-2.1.49-3.amzn2023.0.3.noarch                   4/12
  Installing       : httpd-tools-2.4.54-3.amzn2023.0.4.x86_                 5/12
  Installing       : generic-logos-httpd-18.0.0-12.amzn2023                 6/12
  Running scriptlet: httpd-filesystem-2.4.54-3.amzn2023.0.4                 7/12
  Installing       : httpd-filesystem-2.4.54-3.amzn2023.0.4                 7/12
  Installing       : httpd-core-2.4.54-3.amzn2023.0.4.x86_6                 8/12
  Installing       : mod_http2-1.15.24-1.amzn2023.0.3.x86_6                 9/12
  Installing       : libbrotli-1.0.9-4.amzn2023.0.2.x86_64                 10/12
  Installing       : mod_lua-2.4.54-3.amzn2023.0.4.x86_64                  11/12
  Installing       : httpd-2.4.54-3.amzn2023.0.4.x86_64                    12/12
  Running scriptlet: httpd-2.4.54-3.amzn2023.0.4.x86_64                    12/12
  Verifying        : apr-1.7.2-2.amzn2023.0.2.x86_64                        1/12
  Verifying        : apr-util-openssl-1.6.3-1.amzn2023.0.1.                 2/12
  Verifying        : httpd-core-2.4.54-3.amzn2023.0.4.x86_6                 3/12
  Verifying        : mod_http2-1.15.24-1.amzn2023.0.3.x86_6                 4/12
```

```
  Verifying        : apr-util-1.6.3-1.amzn2023.0.1.x86_64      5/12
  Verifying        : mod_lua-2.4.54-3.amzn2023.0.4.x86_64      6/12
  Verifying        : libbrotli-1.0.9-4.amzn2023.0.2.x86_64     7/12
  Verifying        : httpd-2.4.54-3.amzn2023.0.4.x86_64        8/12
  Verifying        : httpd-tools-2.4.54-3.amzn2023.0.4.x86_    9/12
  Verifying        : mailcap-2.1.49-3.amzn2023.0.3.noarch     10/12
  Verifying        : httpd-filesystem-2.4.54-3.amzn2023.0.4   11/12
  Verifying        : generic-logos-httpd-18.0.0-12.amzn2023   12/12

Installed:
  apr-1.7.2-2.amzn2023.0.2.x86_64
  apr-util-1.6.3-1.amzn2023.0.1.x86_64
  apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
  httpd-2.4.54-3.amzn2023.0.4.x86_64
  httpd-core-2.4.54-3.amzn2023.0.4.x86_64
  httpd-filesystem-2.4.54-3.amzn2023.0.4.noarch
  httpd-tools-2.4.54-3.amzn2023.0.4.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64
  mailcap-2.1.49-3.amzn2023.0.3.noarch
  mod_http2-1.15.24-1.amzn2023.0.3.x86_64
  mod_lua-2.4.54-3.amzn2023.0.4.x86_64

Complete!
```

Adicionar, habilitar ou desabilitar novos repositórios

> **Warning**
>
> Adicione somente repositórios projetados para serem usados com AL2 023.
> Embora os repositórios projetados para outras distribuições possam funcionar atualmente, não há garantia de que continuarão funcionando com qualquer atualização de pacote no AL2 023 ou com o repositório não projetado para uso com o 023. AL2

Para instalar um pacote de um repositório diferente dos repositórios padrão do Amazon Linux, você precisará configurar o sistema de gerenciamento de DNF pacotes para saber onde está o repositório

Para falar `dnf` sobre um repositório de pacotes, adicione as informações do repositório a um arquivo de configuração desse repositório no diretório. `/etc/yum.repos.d/` Muitos repositórios

de terceiros fornecem o conteúdo do arquivo de configuração ou um pacote instalável que inclui o arquivo de configuração.

> **Note**
>
> Embora os repositórios possam ser configurados diretamente no `/etc/dnf/dnf.conf` arquivo, isso não é recomendado. É recomendável que cada repositório seja configurado em seu próprio arquivo no`/etc/yum.repos.d/`.

Para descobrir quais repositórios estão atualmente habilitados, você pode executar o seguinte comando:

```
$ dnf repolist all --verbose
Loaded plugins: builddep, changelog, config-manager, copr, debug, debuginfo-install,
 download, generate_completion_cache, groups-manager, needs-restarting, playground,
 release-notification, repoclosure, repodiff, repograph, repomanage, reposync,
 supportinfo
DNF version: 4.12.0
cachedir: /var/cache/dnf
Last metadata expiration check: 0:00:02 ago on Wed Mar  1 23:40:15 2023.
Repo-id            : amazonlinux
Repo-name          : Amazon Linux 2023 repository
Repo-status        : enabled
Repo-revision      : 1677203368
Repo-updated       : Fri Feb 24 01:49:28 2023
Repo-pkgs          : 12632
Repo-available-pkgs: 12632
Repo-size          : 12 G
Repo-mirrors       : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/x86_64/mirror.list
Repo-baseurl       : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/guids/
cf9296325a6c46ff40c775a8e2d632c4c3fd9d9164014ce3304715d61b90ca8e/x86_64/
                   : (0 more)
Repo-expire        : 172800 second(s) (last: Wed Mar  1 23:40:15
                   : 2023)
Repo-filename      : /etc/yum.repos.d/amazonlinux.repo

Repo-id            : amazonlinux-debuginfo
Repo-name          : Amazon Linux 2023 repository - Debug
Repo-status        : disabled
```

```
Repo-mirrors       : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/debuginfo/x86_64/mirror.list
Repo-expire        : 21600 second(s) (last: unknown)
Repo-filename      : /etc/yum.repos.d/amazonlinux.repo

Repo-id            : amazonlinux-source
Repo-name          : Amazon Linux 2023 repository - Source packages
Repo-status        : disabled
Repo-mirrors       : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/SRPMS/mirror.list
Repo-expire        : 21600 second(s) (last: unknown)
Repo-filename      : /etc/yum.repos.d/amazonlinux.repo

Repo-id            : kernel-livepatch
Repo-name          : Amazon Linux 2023 Kernel Livepatch repository
Repo-status        : disabled
Repo-mirrors       : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/kernel-livepatch/mirrors/al2023/x86_64/mirror.list
Repo-expire        : 172800 second(s) (last: unknown)
Repo-filename      : /etc/yum.repos.d/kernel-livepatch.repo

Repo-id            : kernel-livepatch-source
Repo-name          : Amazon Linux 2023 Kernel Livepatch repository -
                   : Source packages
Repo-status        : disabled
Repo-mirrors       : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/kernel-livepatch/mirrors/al2023/SRPMS/mirror.list
Repo-expire        : 21600 second(s) (last: unknown)
Repo-filename      : /etc/yum.repos.d/kernel-livepatch.repo
Total packages: 12632
```

> **Note**
>
> Se você não adicionar o sinalizador de opção `--verbose`, a saída incluirá somente as informações de `Repo-id`, `Repo-name` e `Repo-status`.

Para adicionar um repositório **`yum`** ao diretório **`/etc/yum.repos.d`**:

1. Encontre a localização do arquivo `.repo`. Neste exemplo, o arquivo `.repo` está em *`https://www.example.com/repository.repo`*.
2. Adicione um repositório com o comando `dnf config-manager`.

```
$ sudo dnf config-manager --add-repo https://www.example.com/repository.repo
Loaded plugins: priorities, update-motd, upgrade-helper
adding repo from: https://www.example.com/repository.repo
grabbing file https://www.example.com/repository.repo to /etc/
yum.repos.d/repository.repo
repository.repo                                          | 4.0 kB     00:00
repo saved to /etc/yum.repos.d/repository.repo
```

Após instalar um repositório, é necessário habilitá-lo como descrito no próximo procedimento.

Para habilitar um yum repositório`/etc/yum.repos.d`, use o `dnf config-manager` comando com a `--enable` bandeira e o *`repository`* nome.

```
$ sudo dnf config-manager --enable repository
```

> **Note**
>
> Para desativar um repositório, use a mesma sintaxe de comando, mas substitua `--enable` por `--disable` no comando.

Adicionando repositórios com cloud-init

Além de adicionar um repositório usando o método anterior, você também pode adicionar um novo repositório usando a estrutura de `cloud-init`.

Para adicionar um novo repositório de pacotes, recomendamos o uso do modelo a seguir. Considere salvar esse arquivo localmente.

```
#cloud-config
yum_repos:
  repository.repo:
    baseurl: https://www.example.com/
    enabled: true
    gpgcheck: true
    gpgkey: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EXAMPLE
    name: Example Repository
```

> **Note**
>
> Uma vantagem de usar `cloud-init` é que você pode adicionar uma seção de `packages:` ao seu arquivo de configuração. Nesta seção, você pode incluir os nomes dos pacotes que você deseja instalar. Você pode instalar pacotes do repositório padrão ou do novo repositório que você adicionou ao arquivo `cloud-config`.
> Para obter informações mais específicas sobre a estrutura do arquivo YAML, consulte Adicionar um repositório YUM na Documentação do *`cloud-init`*.

Depois de configurar o arquivo no formato YAML, você pode executá-lo na estrutura do `cloud-init` na AWS CLI. Certifique-se de incluir a opção `--userdata` e o nome do arquivo `.yml` para chamar as operações desejadas.

```
$ aws ec2 run-instances \
  --image-id \
    resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86_64 \
  --instance-type m5.xlarge \
  --region us-east-1 \
  --key-name aws-key-us-east-1 \
  --security-group-ids sg-004a7650 \
  --user-data file://cloud-config.yml
```

Atualização do Kernel Live em 023 AL2

Você pode usar o Kernel Live Patching for AL2 0.23 para aplicar vulnerabilidades de segurança específicas e patches de bugs críticos a um kernel Linux em execução sem reinicializar ou interromper os aplicativos em execução. Além disso, o Kernel Live Patching pode ajudar a melhorar a disponibilidade do seu aplicativo enquanto aplica essas correções até que o sistema possa ser reinicializado.

AWS lança dois tipos de patches ativos do kernel para AL2 023:

- Security updates (Atualizações de segurança): contêm atualizações para vulnerabilidades e exposições comuns (CVEs) do Linux. Normalmente, essas atualizações são classificadas como importantes ou críticas de acordo com as classificações do Boletim de segurança do Amazon Linux. Geralmente, elas são mapeadas com uma pontuação 7 ou maior do Common Vulnerability Scoring System (CVSS – Sistema de pontuação de vulnerabilidades comuns). Em alguns casos,

AWS pode fornecer atualizações antes que um CVE seja atribuído. Nesses casos, os patches podem aparecer como correções de erros.

- Correções de bugs — Inclua correções para bugs críticos e problemas de estabilidade que não estão associados CVEs a.

AWS fornece patches ativos do kernel para uma versão do kernel AL2 023 por até 3 meses após seu lançamento. Após esse período, é necessário fazer a atualização para uma versão posterior do kernel para continuar a receber patches ao vivo do kernel.

AL2Os patches ativos do kernel 023 são disponibilizados como pacotes RPM assinados nos repositórios AL2 023 existentes. Os patches podem ser instalados em instâncias individuais usando fluxos de trabalho de gerenciamento de pacotes DNF existentes. Ou eles podem ser instalados em um grupo de instâncias gerenciadas usando o AWS Systems Manager.

O Kernel Live Patching em AL2 023 é fornecido sem custo adicional.

Tópicos

Limitações

Ao aplicar um patch ao vivo no kernel, você não pode executar a hibernação, usar ferramentas avançadas de depuração (como `SystemTap`, `kprobes` e ferramentas baseadas em `eBPF`) ou acessar arquivos de saída `ftrace` usados pela infraestrutura do Kernel Live Patching.

> **Note**
>
> Devido a limitações técnicas, alguns problemas não podem ser resolvidos com patches ativos. Por causa disso, essas correções não serão enviadas no pacote de patch ativo do kernel, mas somente na atualização do pacote nativo do kernel. Você pode instalar o pacote nativo do kernel, atualizar e reinicializar o sistema para ativar os patches normalmente.

Configurações e pré-requisitos compatíveis

O Kernel Live Patching é compatível com EC2 instâncias da Amazon e máquinas virtuais locais que executam 023. AL2

Para usar o Kernel Live Patching em AL2 023, você deve usar o seguinte:

- Um `x86_64` de 64 bits ou arquitetura `ARM64`
- Versão `6.1` do Kernel

Requisitos de política

Para baixar pacotes de AL2 023 repositórios, a Amazon EC2 precisa acessar os buckets Amazon S3 de propriedade do serviço. Se você estiver usando um endpoint Amazon Virtual Private Cloud (VPC) para o Amazon S3 em seu ambiente, certifique-se de que sua política de endpoint de VPC permita acesso a esses buckets públicos. A tabela a seguir descreve o bucket do Amazon S3 que a Amazon EC2 pode precisar acessar para o Kernel Live Patching.

ARN do bucket do S3	Descrição
arn:aws:s3: ::al2023-repos- -de612dc2/* *region*	Bucket Amazon S3 contendo 023 repositórios AL2

Trabalhar com o Kernel Live Patching

Você pode habilitar e usar o Kernel Live Patching em instâncias individuais usando a linha de comando na própria instância. Como alternativa, você pode habilitar e usar o Kernel Live Patching em um grupo de instâncias gerenciadas usando o AWS Systems Manager.

As seções a seguir explicam como habilitar e usar o Kernel Live Patching em instâncias individuais usando a linha de comando.

Para obter mais informações sobre como ativar e usar o Kernel Live Patching em um grupo de instâncias gerenciadas, consulte Usar o Kernel Live Patching em AL2 023 instâncias no Guia do usuário.AWS Systems Manager

Tópicos

- Habilitar o Kernel Live Patching

- Visualizar os patches ao vivo do kernel disponíveis
- Aplicar patches ao vivo do kernel
- Visualizar os patches ao vivo do kernel aplicados
- Desabilitar o Kernel Live Patching

Habilitar o Kernel Live Patching

O Kernel Live Patching está desativado por padrão em 023. AL2 Para usar patches ao vivo, você deve instalar o plug-in DNF para patches ao vivo do kernel e ativar a funcionalidade de patching ao vivo.

Como habilitar o Kernel Live Patching

1. Os patches ativos do kernel estão disponíveis para AL2 0.23 com a versão do kernel. `6.1` Para verificar a versão do kernel, execute o comando a seguir.

   ```
   $ sudo dnf list kernel
   ```

2. Instale o plug-in DNF para o Kernel Live Patching.

   ```
   $ sudo dnf install -y kpatch-dnf
   ```

3. Habilite o plug-in DNF para o Kernel Live Patching.

   ```
   $ sudo dnf kernel-livepatch -y auto
   ```

   Este comando também instala a versão mais recente de RPM do patch ao vivo do kernel a partir dos repositórios configurados.

4. Para confirmar se o plug-in DNF para a aplicação de patches ao vivo no kernel foi instalado com êxito, execute o comando a seguir.

   Quando você habilita o Kernel Live Patching, um RPM de patch ao vivo do kernel vazio é aplicado automaticamente. Se o Kernel Live Patching foi habilitado com sucesso, esse comando retornará uma lista que inclui o RPM do patch ativo inicial vazio do kernel (e outro RPM configurando o repositório DNF contendo os livepatches).

   ```
   $ sudo rpm -qa | grep kernel-livepatch
   kernel-livepatch-repo-s3-2023.7.20250428-0.amzn2023.noarch
   ```

```
kernel-livepatch-6.1.134-150.224-1.0-0.amzn2023.x86_64
```

5. Instale o pacote kpatch.

```
$ sudo dnf install -y kpatch-runtime
```

6. Atualize o serviço kpatch, caso tenha sido instalado anteriormente.

```
$ sudo dnf upgrade kpatch-runtime
```

7. Inicie o serviço kpatch. Este serviço carrega todos os patches ao vivo do kernel durante ou após a inicialização.

```
$ sudo systemctl enable kpatch.service && sudo systemctl start kpatch.service
```

Visualizar os patches ao vivo do kernel disponíveis

Os alertas de segurança do Amazon Linux são publicados no Centro de segurança do Amazon Linux. Para obter mais informações sobre os alertas de segurança AL2 023, incluindo alertas para patches ativos do kernel, consulte o Amazon Linux Security Center. Os patches ao vivo do kernel são prefixados com `ALASLIVEPATCH`. O Centro de segurança do Amazon Linux pode não listar patches ao vivo do kernel que resolvam erros.

Você também pode descobrir os patches ativos do kernel disponíveis para recomendações e CVEs usar a linha de comando.

Como listar todos os patches ao vivo do kernel disponíveis para recomendações

Use o seguinte comando.

```
$ sudo dnf updateinfo list
Last metadata expiration check: 1:06:23 ago on Mon 13 Feb 2023 09:28:19 PM UTC.
ALAS2LIVEPATCH-2021-123   important/Sec. kernel-
livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
ALAS2LIVEPATCH-2022-124   important/Sec. kernel-
livepatch-6.1.12-17.42-1.0-3.amzn2023.x86_64
```

Para listar todos os patches ativos do kernel disponíveis para CVEs

Use o seguinte comando.

```
$ sudo dnf updateinfo list cves
Last metadata expiration check: 1:07:26 ago on Mon 13 Feb 2023 09:28:19 PM UTC.
CVE-2022-0123    important/Sec. kernel-livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
CVE-2022-3210    important/Sec. kernel-livepatch-6.1.12-17.42-1.0-3.amzn2023.x86_64
```

Aplicar patches ao vivo do kernel

Aplique patches ao vivo do kernel usando o gerenciador de pacotes DNF da mesma maneira que você aplicaria atualizações regulares. O plug-in DNF para Kernel Live Patching gerencia os patches ativos do kernel que estão disponíveis para serem aplicados.

> **Tip**
>
> Recomendamos que você atualize seu kernel regularmente usando o Kernel Live Patching para garantir que ele receba correções de segurança específicas importantes e críticas até que o sistema possa ser reinicializado. Verifique também se correções adicionais foram disponibilizadas para o pacote nativo do kernel que não podem ser implantadas como patches ativos e atualize e reinicie na atualização do kernel nesses casos.

É possível optar por aplicar um patch ao vivo do kernel específico, ou aplicar qualquer patch ao vivo do kernel disponível com suas atualizações de segurança regulares.

Como aplicar um patch ao vivo do kernel específico

1. Obtenha a versão do patch ao vivo do kernel usando um dos comandos descritos em Visualizar os patches ao vivo do kernel disponíveis.
2. Aplique o patch ativo do kernel para seu kernel AL2 023.

   ```
   $ sudo dnf install kernel-livepatch-kernel_version-package_version.amzn2023.x86_64
   ```

   Por exemplo, o comando a seguir aplica um patch ativo do kernel para a versão AL2 0.23 do kernel. `6.1.12-17.42`

   ```
   $ sudo dnf install kernel-livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
   ```

Como aplicar patches ao vivo do kernel disponíveis com as atualizações de segurança regulares

Use o seguinte comando.

```
$ sudo dnf upgrade --security
```

Omita a opção `--security` para incluir correções de erros.

> **Important**
>
> - A versão do kernel não é atualizada após a aplicação de patches ao vivo do kernel. A versão só é atualizada para a nova versão depois da reinicialização da instância.
> - Um kernel AL2 023 recebe patches ativos do kernel por 3 meses. Após esse período, nenhum novo patch ativo do kernel será lançado para essa versão do kernel.
> - Para continuar a receber patches ao vivo do kernel após 3 meses, você deve reinicializar a instância para migrar para a nova versão do kernel. A instância continua recebendo patches ativos do kernel pelos próximos 3 meses após a atualização.
> - Para verificar a janela de suporte para a versão do kernel, execute o seguinte comando.
>
>   ```
>   $ sudo dnf kernel-livepatch support
>   The current version of the Linux kernel you are running will no longer receive
>    live patches after 2025-07-22.
>   ```

Visualizar os patches ao vivo do kernel aplicados

Como visualizar os patches ao vivo do kernel aplicados

Use o seguinte comando.

```
$ sudo kpatch list
Loaded patch modules:
livepatch_CVE_2022_36946 [enabled]

Installed patch modules:
livepatch_CVE_2022_36946 (6.1.57-29.131.amzn2023.x86_64)
livepatch_CVE_2022_36946 (6.1.57-30.131.amzn2023.x86_64)
```

O comando retornará uma lista dos patches ao vivo do kernel de atualização de segurança carregados e instalados. A seguir está um exemplo de saída.

> **Note**
>
> Um único patch ao vivo do kernel pode incluir e instalar vários patches ao vivo.

Desabilitar o Kernel Live Patching

Se não precisar mais usar o Kernel Live Patching, é possível desabilitá-lo a qualquer momento.

- Desative o uso de livepatches:
  1. Desabilite o plug-in:

     ```
     $ sudo dnf kernel-livepatch manual
     ```

  2. Desative o kpatch serviço:

     ```
     $ sudo systemctl disable --now kpatch.service
     ```

- Remova totalmente o livepatch ferramentas:
  1. Remover o plug-in de:

     ```
     $ sudo dnf remove kpatch-dnf
     ```

  2. Remover kpatch-runtime:

     ```
     $ sudo dnf remove kpatch-runtime
     ```

  3. Remova qualquer um instalado livepatches:

     ```
     $ sudo dnf remove kernel-livepatch\*
     ```

Atualizando o kernel Linux em 023 AL2

Tópicos

- Versões do kernel Linux em 023 AL2
- Atualizando AL2 0.23 para o kernel 6.12
- AL2023 kernels - Perguntas frequentes

Versões do kernel Linux em 023 AL2

AL2O 023 inclui regularmente novas versões do kernel com base nas versões Long-Term Support (LTS) do kernel Linux.

AL2023 foi lançado originalmente em março de 2023 com o kernel 6.1.

Em abril de 2025, AL2 023 adicionou suporte para o kernel Linux 6.12. Esse kernel adicionou novos recursos, incluindo agendamento EEVDF, suporte a E/S de passagem do FUSE, uma nova API Futex e melhorias no eBPF. O Kernel 6.12 também permite que um programa de espaço de usuário se proteja em tempo de execução usando pilhas de sombras do espaço do usuário e vedação de memória.

Atualizando AL2 0.23 para o kernel 6.12

Você pode executar o AL2 023 com o kernel 6.12 selecionando uma AMI com o kernel 6.12 pré-instalado ou atualizando uma instância 023 existente. AL2 EC2

Executando uma AL2 AMI do kernel 0.23 6.12

Você pode optar por executar uma AMI AL2 023 com o kernel 6.12 pré-instalado por meio do AWS Console ou consultando o SSM para parâmetros específicos. As chaves SSM a serem consultadas começam com, `/aws/service/ami-amazon-linux-latest/` seguidas por uma das

- `al2023-ami-kernel-6.12-arm64` para a arquitetura arm64
- `al2023-ami-minimal-kernel-6.12-arm64` para arquitetura arm64 (AMI mínima)
- `al2023-ami-kernel-6.12-x86_64` para a arquitetura x86_64
- `al2023-ami-minimal-kernel-6.12-x86_64` para a arquitetura x86_64 (AMI mínima)

Consulte Iniciando AL2 023 usando o parâmetro SSM e AWS CLI para obter detalhes sobre a seleção de AL2 023 AMIs.

Atualização de uma instância AL2 023 para o kernel 6.12

Você pode atualizar localmente uma instância AL2 023 em execução para o kernel 6.12 com as seguintes etapas:

1. Instale o pacote `kernel6.12`:

```
$ sudo dnf install -y kernel6.12
```

2. Obtenha a versão mais recente do `kernel6.12` pacote:

```
$ version=$(rpm -q --qf '%{version}-%{release}.%{arch}\n' kernel6.12 | sort -V |
 tail -1)
```

3. Torne o novo `kernel6.12` seu kernel padrão:

```
$ sudo grubby --set-default "/boot/vmlinuz-$version"
```

4. Reinicie seu sistema:

```
$ sudo reboot
```

5. (Opcional) Desinstale o kernel 6.1:

```
$ sudo dnf remove -y kernel
```

Fazendo o downgrade do kernel 6.12 para o kernel 6.1

Se em algum momento você precisar fazer o downgrade para o kernel 6.1, use as seguintes etapas:

1. Certifique-se de instalar o `kernel` pacote:

```
$ sudo dnf install -y kernel
```

2. Obtenha a versão mais recente do `kernel` pacote:

```
$ version=$(rpm -q --qf '%{version}-%{release}.%{arch}\n' kernel | sort -V | tail
 -1)
```

3. Faça do kernel 6.1 seu kernel padrão:

```
$ sudo grubby --set-default "/boot/vmlinuz-$version"
```

4. Reinicie seu sistema:

```
$ sudo reboot
```

5. (Opcional) Desinstale o kernel 6.12:

```
$ sudo dnf remove -y kernel6.12
```

AL2023 kernels - Perguntas frequentes

1. Preciso reinicializar após uma atualização do kernel?

Cada alteração no kernel em execução requer uma reinicialização.

2. Como faço para manter os kernels up-to-date em várias instâncias?

O Amazon Linux não fornece recursos para gerenciar frotas de instâncias. Recomendamos que você corrija grandes frotas usando ferramentas como o AWS Systems Manager.

3. Como faço para verificar qual versão do kernel estou executando no momento?

Execute esse comando na sua instância AL2 023:

```
$ uname -r
```

4. Como faço para instalar cabeçalhos do kernel, pacotes de desenvolvimento e módulos extras para o kernel 6.12?

Por favor, execute:

```
$ sudo dnf install -y kernel6.12-modules-extra-$(uname -r) kernel-headers-$(uname -r)
 kernel-devel-$(uname -r)
```

5. Como seleciono a versão correta do **`perf`** para o meu kernel?

`perf`Os recursos do estão fortemente vinculados à versão do kernel que você está executando. Nós fornecemos pacotes `perf` para o kernel 6.1 e `perf6.12` para o kernel 6.12. Se você `perf` instalou e gostaria de mudar para a versão 6.12 do kernel, execute:

```
$ dnf -y swap perf perf6.12
```

Introdução aos tempos de execução de programação em 023 AL2

AL2O 023 fornece versões diferentes de alguns tempos de execução de linguagem. Trabalhamos com projetos upstream que oferecem suporte a várias versões ao mesmo tempo. Encontre informações sobre como instalar e gerenciar esses pacotes com versão por nome usando o comando `dnf` para pesquisar e instalar esses pacotes.

Os tópicos a seguir descrevem como cada ecossistema de linguagem existe em AL2 023.

Tópicos

C, C++ e Fortran em AL2 023

AL2023 inclui a coleção de compiladores GNU (GCC) e o Clang front-end para LLVM (Máquina virtual de baixo nível).

A versão principal do GCC permanecerá constante durante toda a vida útil de AL2 023. Versões menores trazem correções de bugs e podem ser incluídas nas versões AL2 023. Outras correções de bugs, desempenho e segurança podem ser transferidas para a versão principal do GCC que será lançado em AL2 023.

AL2023 inclui a versão 11 do GCC com os front-ends C (`gcc`), C++ (`g++`) e Fortran (`gfortran`).

AL2023 não habilita o Ada (`gnat`), Go (`gcc-go`), interfaces Objective-C ou Objective-C++.

Os sinalizadores padrão do compilador com os quais o AL2 023 RPMs é construído incluem sinalizadores de otimização e fortalecimento. Para criar seu próprio código com o GCC, recomendamos que você inclua sinalizadores de otimização e fortalecimento.

> **Note**
>
> Quando `gcc --version` é invocado, uma string de versão como `gcc (GCC) 11.3.1 20221121 (Red Hat 11.3.1-4)` é exibida. `Red Hat` refere-se à filial do fornecedor do GCC na qual o pacote Amazon Linux GCC se baseia. De acordo com o URL do relatório de bugs exibido por`gcc --help`, todos os relatórios de bugs e solicitações de suporte devem ser direcionados para o Amazon Linux.
> Para obter mais informações sobre algumas das mudanças de longo prazo nessa ramificação do fornecedor, como a `__GNUC_RH_RELEASE__` macro, consulte Fontes de pacotes do Fedora.

Para obter mais informações sobre o conjunto de ferramentas principal, consultePacotes principais do conjunto de ferramentas glibc, gcc, binutils.

Para obter mais informações sobre o AL2 023 e sua relação com outras distribuições Linux, consulte. Relacionamento com o Fedora

Para obter mais informações sobre a alteração do tripleto do compilador em AL2 023 em comparação com, consulte. AL2 Compilador Triplet

Go em AL2 023

Talvez você queira criar seu próprio código escrito em Gono Amazon Linux, e talvez queira usar um conjunto de ferramentas fornecido com o AL2 023. Semelhante a AL2, AL2 023 atualizará o Go conjunto de ferramentas durante toda a vida útil do sistema operacional. Isso pode ser em resposta a qualquer CVE no conjunto de ferramentas que enviamos ou como parte de uma versão trimestral.

Go é uma linguagem que se move relativamente rápido. Pode haver uma situação em que os aplicativos existentes sejam escritos em Go tem que se adaptar às novas versões do Go conjunto de ferramentas. Para obter mais informações sobre Go, veja Go 1 e o futuro do Go Programas.

Embora o AL2 023 incorpore novas versões do Go cadeia de ferramentas durante sua vida útil, isso não estará em sintonia com o upstream Go lançamentos. Portanto, usando o Go o conjunto de

ferramentas fornecido em AL2 023 pode não ser adequado se você quiser construir Go codifique usando recursos de ponta do Go linguagem e biblioteca padrão.

Durante a vida útil do AL2 023, as versões anteriores do pacote não são removidas dos repositórios. Se um anterior Go o conjunto de ferramentas é necessário, você pode optar por renunciar às correções de bugs e segurança mais recentes Go cadeias de ferramentas e instale uma versão anterior dos repositórios usando os mesmos mecanismos disponíveis para qualquer RPM.

Se você quiser construir o seu Go código em AL2 023, você pode usar o Go conjunto de ferramentas incluído em AL2 023 com o conhecimento de que esse conjunto de ferramentas pode avançar durante a vida útil de 023. AL2

AL2023 Funções Lambda escritas em Go

Como Go compila em código nativo, o Lambda trata Go como um tempo de execução personalizado. Você pode usar o `provided.al2023` tempo de execução para implantar Go funciona em AL2 023 para Lambda.

Para obter mais informações, consulte Criando funções Lambda com Go no Guia do desenvolvedor do AWS Lambda .

Javaem AL2 023

AL2O 023 fornece várias versões do Amazon Corretto para Java suportar cargas de trabalho baseadas. Todos os pacotes Java baseados incluídos no AL2 023 são construídos comAmazon Corretto 17.

Corretto é uma versão do Open Java Development Kit (OpenJDK) com suporte de longo prazo da. Amazon Corretto é certificado usando o Java Technical Compatibility Kit (TCK) para garantir que ele atenda ao padrão Java SE e esteja disponível em Linux, e. Windows macOS

Há um pacote Amazon Corretto disponível para cada Corretto 1.8.0, Corretto 11 e Corretto 17.

Cada versão do Corretto em AL2 023 é suportada pelo mesmo período de tempo que a versão do Corretto, ou até o final da vida útil de AL2 023, o que ocorrer primeiro. Para obter mais informações, consulte as declarações de suporte do pacote Amazon Linux e o Amazon Corretto FAQs.

NodeJS em AL2 023

NodeJSem AL2 023 é representado pelas versões 18, 20 e 22. Eles têm namespaces e podem ser instalados simultaneamente no mesmo sistema. NodeJS é distribuído como vários pacotes que incluem o node, a ferramenta npm de uma versão compatível com ele, documentação, bibliotecas, etc. Por exemplo, para NodeJS 18, node e npm são fornecidos pelos `nodejs-npm` pacotes `nodejs` e. No entanto, todas as versões seguintes do NodeJS têm nomes de pacotes com namespaces que começam com. `nodejs{MAJOR_VERSION}` Por exemplo, NodeJS 20, vem com node e npm empacotados como `nodejs20` e `nodejs20-npm` respectivamente.

Para permitir a instalação simultânea de diferentes versões principais do NodeJS, os pacotes são fornecidos com executáveis, módulos e outros arquivos com namespaces para evitar sobreposições e conflitos no sistema de arquivos. Por exemplo, o executável do nó é nomeado `/usr/bin/node-{MAJOR_VERSION}` e o executável npm é nomeado. `/usr/bin/npm-{MAJOR_VERSION}` No entanto, só pode haver um `/usr/bin/node` e um `/usr/bin/npm` no sistema em execução. Esses executáveis são nomes virtuais (links simbólicos) e apontam para os executáveis reais da versão atualmente ativa do NodeJS. Isso é conseguido usando o sistema de alternativas.

O uso de alternativas permite que você use um único comando para selecionar quais NodeJS arquivos de configuração da versão, binários (como `node` enpm) e módulos instalados globalmente são usados. Por padrão, as alternativas são configuradas para estar no modo automático, que usa prioridades para selecionar a versão atualmente ativa do NodeJS. No entanto, você pode alternar entre as versões instaladas a qualquer momento executando`alternatives --config node`. Atualmente, todas as versões suportadas do NodeJS têm a mesma prioridade.

Alguns comandos alternativos úteis:

1. Verifique o que está configurado nas alternativas

   ```
   alternatives --list
   ```

2. Verifique a configuração atual do nó

   ```
   alternatives --display node
   ```

3. Altere interativamente o NodeJS version

   ```
   alternatives --config node
   ```

4. Mude para o modo manual e selecione uma versão específica

```
alternatives --set node /usr/bin/node-{MAJOR_VERSION}
```

5. Volte para o modo de seleção automática de versão

```
alternatives --auto node
```

Perl em AL2 023

AL2023 fornece a versão 5.32 do Perllinguagem de programação.

Apesar Perl proporcionou um alto grau de compatibilidade linguística como parte do Perl 5 lançamentos nas últimas décadas, não se espera que o Amazon Linux saia Perl 5.32 durante a versão AL2 023. O Amazon Linux continuará com o patch de segurança Perl durante a vida útil de AL2 023, de acordo com nossas declarações de suporte de pacotes.

Perl módulos em AL2 023

Diversos Perl os módulos são empacotados como RPMs em AL2 023. Embora existam muitos Perl módulos disponíveis como RPMs, o Amazon Linux não visa empacotar todos os módulos possíveis Perl módulo. Módulos empacotados RPMs de acordo com os pacotes RPM de outros sistemas operacionais, portanto, o Amazon Linux priorizará esses patches de segurança em vez de atualizações puras de recursos.

AL2023 também inclui `CPAN` para que Perl os desenvolvedores podem usar o gerenciador de pacotes idiomático para Perl módulos.

PHP em AL2 023

AL2023 atualmente fornece o PHPlinguagem de programação, versões 8.1, 8.2, 8.3 e 8.4. Cada versão é suportada pelo mesmo período de tempo que o upstream PHP. Para obter mais informações, consulte Declarações de suporte do Package.

Migrando do antigo PHP versões

O rio acima PHP a comunidade reuniu uma documentação abrangente de migração para mover:

- de PHP 8.3.x a PHP 8.4.x

- de PHP 8.2.x a PHP 8.3.x
- de PHP 8.1.x a PHP 8.2.x
- de PHP 8.0.x a PHP 8.1.x

AL2 inclui PHP 8.0, 8.1 e 8.2 para `amazon-linux-extras` permitir um caminho de atualização fácil para AL2 023.

Migrar a partir de PHP Versões 7.x

> **Note**
>
> A PHPO projeto mantém uma lista e um cronograma das versões suportadas, bem como uma lista de ramificações não suportadas.
> Quando o AL2 023 foi lançado, todas as versões 7.x e 5.x do PHPnão foram apoiados pelo PHP comunidade, e não foram incluídos como opções em AL2 023.

O rio acima PHP a comunidade reuniu uma documentação abrangente de migração para a mudança para PHP 8,0 de PHP 7.4. Combinado com a documentação mencionada na seção anterior sobre migração para PHP 8.1 e PHP 8.2, você pode migrar seu PHP aplicação baseada em moderna PHP.

> **Note**
>
> AL2 inclui PHP 7,1, 7,2, 7,3 e 7,4 pol. `amazon-linux-extras` É importante observar que todos esses extras têm end-of-life ou não garantia de receber mais atualizações de segurança.

PHP módulos em AL2 023

AL2023 inclui muitos PHP módulos que estão incluídos no PHP Núcleo. AL2023 não visa incluir todos os pacotes no PHP Biblioteca Comunitária de Extensão (PECL).

Python em AL2 023

AL2023 removido Python 2.7 e quaisquer componentes que exijam Python agora estão escritos para trabalhar com Python 3.

AL2023 marcas Python 3 disponível `/usr/bin/python3` para manter a compatibilidade com o código do cliente, bem como com o código Python enviado com AL2 023, isso permanecerá como Python 3.9 por toda a vida útil de AL2 023.

A versão do python para a qual `/usr/bin/python3` aponta é considerada o sistema Python e, AL2 para 0.23, isso é Python 3.9.

Versões mais recentes do Python, por exemplo, Python 3.11, são disponibilizados como pacotes na versão AL2 023 e são suportados durante toda a vida útil das versões upstream. Para obter informações sobre por quanto tempo o Python 3.11 é suportado, consulte Python 3.11.

Várias versões do Python pode ser instalado simultaneamente no AL2 023. Embora `/usr/bin/python3` sempre seja Python 3.9, cada versão do Python tem namespace e pode ser encontrada pelo número da versão. Por exemplo, se `python3.11` estiver instalado, `/usr/bin/python3.11` existirá ao lado `/usr/bin/python3.9` e o symlink `/usr/bin/python3` apontará para `/usr/bin/python3.9`.

> **Note**
>
> Não altere o que o `/usr/bin/python3` link simbólico aponta, pois isso pode quebrar a funcionalidade principal do AL2 023.

Python módulos em AL2 023

Vários Python os módulos são empacotados como RPMs em AL2 023. Normalmente, RPMs para Python os módulos serão criados visando apenas a versão do sistema do Python.

Rust em AL2 023

Talvez você queira construir seu código escrito em Rustno Amazon Linux e talvez queira usar um conjunto de ferramentas fornecido com o AL2 023.

Semelhante a AL2, AL2 023 atualizará o Rust conjunto de ferramentas durante toda a vida útil do sistema operacional. Isso pode ser em resposta a qualquer CVE no conjunto de ferramentas que enviamos ou como parte de uma versão trimestral.

Rust é uma linguagem relativamente rápida, com novos lançamentos em uma cadência de aproximadamente seis semanas. Essas versões podem adicionar um novo idioma ou recursos de biblioteca padrão. Embora o AL2 023 incorpore novas versões do Rust cadeia de ferramentas durante sua vida útil, isso não estará em sintonia com o upstream Rust lançamentos. Portanto, usando o Rust o conjunto de ferramentas fornecido em AL2 023 pode não ser adequado se você quiser construir Rust codifique usando recursos de ponta do Rust idioma.

Durante a vida útil do AL2 023, as versões antigas do pacote não são removidas dos repositórios. Se um mais velho Rust o conjunto de ferramentas é necessário, você pode optar por renunciar às correções de bugs e segurança mais recentes Rust cadeias de ferramentas e instale uma versão mais antiga dos repositórios usando os mesmos mecanismos disponíveis para qualquer RPM.

Se você quiser construir o seu próprio Rust código em AL2 023, você pode usar o Rust conjunto de ferramentas incluído em AL2 023 com o conhecimento de que esse conjunto de ferramentas pode avançar durante a vida útil de 023. AL2

AL2023 Funções Lambda escritas em Rust

Porque Rust compila em código nativo, o Lambda trata Rust como um tempo de execução personalizado. Você pode usar o `provided.al2023` tempo de execução para implantar Rust funciona em AL2 023 para Lambda.

Para obter mais informações, consulte Criando funções Lambda com Rust no Guia do desenvolvedor do AWS Lambda .

AL2023 Usuários e grupos reservados

AL2O 023 pré-aloca determinados usuários e grupos durante o provisionamento da imagem e durante a instalação de determinados pacotes. Os usuários, grupos e seus associados UIDs GIDs estão listados aqui para evitar conflitos.

Tópicos

Lista de AL2 023 usuários reservados

Nome do usuário	UID
raiz	0
bin	1
daemon	2
adm	3
lp	4
sincronização	5
shutdown	6
parar	7
correio	8
operador	11
jogos	12
ftp	14

Nome do usuário	UID
lula	23
nomeado	25
postgres	26
mysql	27
nscd	28
nscd	28
rpcuser	29
rpc	32
mailnull	47
apache	48
smmsp	51
tomcat	53
tapa	55
tss	59
nslcd	65
avahi	70
tcpdump	72
sshd	74
radvd	75
dbus	81

Nome do usuário	UID
postfix	89
pombal	97
estapuro	156
stapsys	157
stapdev	158
avahi-autoipd	170
pulso	171
kit rtkit	172
seixo	179
rede systemd	192
resolução do sistema	193
uuidd	961
servidor stap	962
systemd-journal-remote	963
redis6	970
assinar	971
smtpq	972
smtpd	973
nginx	974
munge	975

Nome do usuário	UID
memcached	976
sphinx	977
haproxy	978
flatpak	979
debuginfod	980
pombal	981
dnsmasq	982
não vinculado	983
clamscan	984
clamilt	985
clamupdate	986
colorido	987
probabilidades	988
aws-kinesis-agent-user	989
saslauth	990
cwagent	991
educado	992
ec2-instance-connect	993
chrony	994
sincronização de horário do sistema	995

Nome do usuário	UID
systemd-coredump	996
libstoragemgmt	997
sala do sistema	999
usuário ec2	1000
ninguém	65534

Listado por nome

Nome do usuário	UID
adm	3
apache	48
avahi	70
avahi-autoipd	170
aws-kinesis-agent-user	989
bin	1
chrony	994
clamilt	985
clamscan	984
clamupdate	986
colorido	987
cwagent	991

Nome do usuário	UID
daemon	2
dbus	81
debuginfod	980
dnsmasq	982
pombal	97
pombal	981
ec2-instance-connect	993
usuário ec2	1000
flatpak	979
ftp	14
jogos	12
parar	7
haproxy	978
tapa	55
libstoragemgmt	997
lp	4
correio	8
mailnull	47
memcached	976
munge	975

Nome do usuário	UID
mysql	27
nomeado	25
nginx	974
ninguém	65534
nscd	28
nscd	28
nslcd	65
probabilidades	988
operador	11
assinar	971
educado	992
postfix	89
postgres	26
pulso	171
radvd	75
redis6	970
raiz	0
rpc	32
rpcuser	29
kit rtkit	172

Nome do usuário	UID
seixo	179
saslauth	990
shutdown	6
smmsp	51
smtpd	973
smtpq	972
sphinx	977
lula	23
sshd	74
servidor stap	962
stapdev	158
stapsys	157
estapuro	156
sincronização	5
systemd-coredump	996
systemd-journal-remote	963
rede systemd	192
sala do sistema	999
resolução do sistema	193
sincronização de horário do sistema	995

Nome do usuário	UID
tcpdump	72
tomcat	53
tss	59
não vinculado	983
uuidd	961

Lista de AL2 023 grupos reservados

Group name	GID
raiz	0
bin	1
daemon	2
diz	3
adm	4
tty	5
disco	6
disco	6
lp	7
mem	8
kmem	9
wheel	10

Group name	GID
cdrom	11
correio	12
correio	12
man	15
dialogar	18
flexível	19
jogos	20
deslocar	21
tombada	22
lula	23
nomeado	25
postgres	26
mysql	27
nscd	28
nscd	28
rpcuser	29
rpc	32
fita	33
utempter	35
kvm	36

Group name	GID
video	39
mailnull	47
apache	48
ftp	50
smmsp	51
tomcat	53
bloquear	54
tapa	55
tss	59
áudio	63
avahi	70
tcpdump	72
sshd	74
radvd	75
saslauth	76
dbus	81
screen	84
webpriv	88
postfix	89
pós-entrega	90

Group name	GID
pombal	97
usuários	100
input	104
renderizar	105
sgx	106
zombam	135
estapuro	156
estapuro	156
stapsys	157
stapsys	157
stapdev	158
stapdev	158
avahi-autoipd	170
pulso	171
kit rtkit	172
seixo	179
periódico systemd	190
rede systemd	192
resolução do sistema	193
convocar	959

Group name	GID
tubarão de arame	960
uuidd	961
servidor stap	962
systemd-journal-remote	963
compartilhamentos de usuários	964
redis6	965
assinar	966
smtpq	967
smtpd	968
nginx	969
munge	970
memcached	971
sphinx	972
rastreamento	973
haproxy	974
flatpak	975
debuginfod	976
pombal	977
dnsmasq	978
não vinculado	979

Group name	GID
clamscan	980
clamilt	981
grupo de vírus	982
grupo de vírus	982
grupo de vírus	982
clamupdate	983
administrador de impressão	984
colorido	985
probabilidades	986
docker	987
aws-kinesis-agent-user	988
cwagent	989
pulse-rt	990
acesso por pulso	991
ec2-instance-connect	993
chrony	994
sincronização de horário do sistema	995
systemd-coredump	996
libstoragemgmt	997
chaves_ssh	998

Group name	GID
sala do sistema	999
usuário ec2	1000
um	1001
educado	920
ninguém	65534

Listado por nome

Group name	GID
adm	4
apache	48
áudio	63
avahi	70
avahi-autoipd	170
aws-kinesis-agent-user	988
bin	1
cdrom	11
chrony	994
clamilt	981
clamscan	980
clamupdate	983

Group name	GID
colorido	985
cwagent	989
daemon	2
dbus	81
debuginfod	976
dialogar	18
disco	6
disco	6
dnsmasq	978
docker	987
pombal	97
pombal	977
ec2-instance-connect	993
usuário ec2	1000
flatpak	975
flexível	19
ftp	50
jogos	20
haproxy	974
input	104

Group name	GID
kmem	9
kvm	36
tapa	55
libstoragemgmt	997
bloquear	54
lp	7
correio	12
correio	12
mailnull	47
man	15
mem	8
memcached	971
zombam	135
munge	970
mysql	27
nomeado	25
um	1001
nginx	969
ninguém	65534
nscd	28

Group name	GID
nscd	28
probabilidades	986
assinar	966
educado	920
pós-entrega	90
postfix	89
postgres	26
administrador de impressão	984
pulso	171
acesso por pulso	991
pulse-rt	990
radvd	75
redis6	965
renderizar	105
raiz	0
rpc	32
rpcuser	29
kit rtkit	172
seixo	179
saslauth	76

Group name	GID
screen	84
sgx	106
deslocar	21
smmsp	51
smtpd	968
smtpq	967
sphinx	972
lula	23
chaves_ssh	998
sshd	74
servidor stap	962
stapdev	158
stapdev	158
stapsys	157
stapsys	157
estapuro	156
estapuro	156
diz	3
systemd-coredump	996
periódico systemd	190

Group name	GID
systemd-journal-remote	963
rede systemd	192
sala do sistema	999
resolução do sistema	193
sincronização de horário do sistema	995
fita	33
tcpdump	72
tomcat	53
rastreamento	973
tss	59
tty	5
não vinculado	979
convocar	959
usuários	100
compartilhamentos de usuários	964
utempter	35
tombada	22
uuidd	961
video	39
grupo de vírus	982

Group name	GID
grupo de vírus	982
grupo de vírus	982
webpriv	88
wheel	10
tubarão de arame	960

Lista de codecs disponíveis em 023 AL2

AL2O 023 fornece uma seleção de codecs multimídia por meio de seus repositórios padrão. Esta página fornece uma visão geral dos codecs e seus casos de uso típicos.

> **Important**
>
> O uso e a distribuição de codecs incluídos no Amazon Linux podem exigir que você obtenha direitos de licença de terceiros, incluindo proprietários ou licenciadores de determinados formatos de áudio e vídeo de terceiros. Você é o único responsável por obter essas licenças e pagar quaisquer royalties ou taxas necessárias.

Codec	Descrição
flac	Um codec de áudio sem perdas gratuito e de código aberto que comprime o áudio sem perder nenhum dado ou qualidade, comumente usado para armazenamento de áudio de alta qualidade
fdk-aac-free	Uma implementação de código aberto do padrão AAC (Advanced Audio Codec), fornecendo compressão de áudio de alta qualidade para MP3 alternativas como streaming ou armazenamento de arquivos
webrtc-audio-processing	Uma biblioteca para processamento de áudio usada no WebRTC (Web Real-Time Communication), oferecendo recursos como supressão de ruído, cancelamento de eco e controle de ganho
opus	Um codec de áudio altamente versátil e eficiente projetado para streaming em tempo real, oferecendo baixa latência e suporte para

Codec	Descrição
	uma ampla variedade de aplicativos de áudio, incluindo VoIP e streaming de música
libsndfile	Uma biblioteca para leitura e gravação de arquivos de áudio em vários formatos (como WAV, AIFF e FLAC), comumente usada em ferramentas de processamento e manipulação de áudio

Segurança e compatibilidade no Amazon Linux 2023

> **Important**
>
> Se você quiser denunciar uma vulnerabilidade ou tiver uma preocupação de segurança em relação a serviços em AWS nuvem ou projetos de código aberto, entre em contato com a AWS Segurança usando a página Relatório de vulnerabilidades

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de uma arquitetura de data center e rede criada para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O modelo de responsabilidade compartilhada descreve isso como segurança da nuvem e segurança na nuvem:

- Segurança da nuvem — AWS é responsável por proteger a infraestrutura que executa AWS os serviços na AWS nuvem. AWS também fornece serviços que você pode usar com segurança. Auditores terceirizados testam e verificam regularmente a eficácia de nossa segurança como parte dos Programas de Conformidade Programas de AWS de . Para saber mais sobre os programas de conformidade que se aplicam ao AL2 023, consulte AWS Serviços no escopo do programa de conformidade AWS .
- Segurança na nuvem - sua responsabilidade é determinada pelo serviço AWS que você usa. Você também é responsável por outros fatores, inclusive a sensibilidade de seus dados, os requisitos da sua empresa, leis e regulamentos aplicáveis.

Tópicos

- AL2023 Endurecimento do kernel
- Inicialização segura UEFI em 023 AL2

Consultorias de segurança do Amazon Linux para 2013 AL2

Embora trabalhemos duro para tornar o Amazon Linux seguro, às vezes haverá problemas de segurança que precisam ser corrigidos. Um aviso é emitido quando uma correção está disponível. O principal local onde publicamos recomendações é o Amazon Linux Security Center (ALAS). Para obter informações, consulte o Amazon Linux Security Center.

> **Important**
>
> Se você quiser denunciar uma vulnerabilidade ou tiver uma preocupação de segurança em relação a serviços em AWS nuvem ou projetos de código aberto, entre em contato com a AWS Segurança usando a página Relatórios de vulnerabilidades

As informações sobre problemas e as atualizações relevantes que afetam o AL2 023 são publicadas pela equipe do Amazon Linux em vários locais. É comum que as ferramentas de segurança busquem informações dessas fontes primárias e apresentem os resultados para você. Dessa forma, você pode não interagir diretamente com as fontes primárias que o Amazon Linux publica, mas sim com a interface fornecida pelas suas ferramentas preferidas, como o Amazon Inspector.

Anúncios do Amazon Linux Security Center

Os anúncios do Amazon Linux são fornecidos para itens que não se encaixam em um aviso. Esta seção contém anúncios sobre o próprio ALAS, junto com informações que não cabem em um comunicado. Para obter mais informações, consulte Anúncios do Amazon Linux Security Center (ALAS).

Por exemplo, o anúncio 2021-001 - Amazon Linux Hotpatch para Apache Log4j se encaixa em um anúncio em vez de em um aviso. Neste anúncio, a Amazon Linux adicionou um pacote para ajudar os clientes a mitigar um problema de segurança em software que não fazia parte do Amazon Linux.

O Amazon Linux Security Center CVE Explorer também foi anunciado nos anúncios do ALAS. Para obter mais informações, consulte Novo site para CVEs.

Perguntas frequentes sobre o Amazon Linux Security Center

Para obter respostas a algumas perguntas frequentes sobre o ALAS e como o Amazon Linux avalia CVEs, consulte Perguntas frequentes do Amazon Linux Security Center (ALAS) (). FAQs

Avisos da ALAS

Um Amazon Linux Advisory contém informações importantes relevantes para os usuários do Amazon Linux, geralmente informações sobre atualizações de segurança. O Amazon Linux Security Center é onde os avisos são visíveis na web. As informações consultivas também fazem parte dos metadados do repositório de pacotes RPM.

Consultorias e repositórios RPM

Um repositório de pacotes do Amazon Linux 2023 pode conter metadados descrevendo zero ou mais atualizações. O `dnf updateinfo` comando tem o nome do nome do arquivo de metadados do repositório que contém essas informações,. `updateinfo.xml` Embora o comando tenha um nome `updateinfo` e o arquivo de metadados se refira a um`update`, todos eles se referem a atualizações de pacotes que fazem parte de um Aviso.

Os Amazon Linux Advisories são publicados no site do Amazon Linux Security Center, junto com as informações presentes nos metadados do repositório RPM aos quais o gerenciador de `dnf` pacotes se refere. Eventualmente, os metadados do site e do repositório são consistentes e pode haver inconsistências nas informações do site e nos metadados do repositório. Isso normalmente ocorre quando uma nova versão do AL2 023 está em processo de lançamento. Há uma atualização para um Aviso após a versão mais recente do AL2 023.

Embora seja comum que um novo aviso seja emitido junto com a atualização do pacote que resolve o problema, esse nem sempre é o caso. Um aviso pode ser criado para um novo problema que é tratado em pacotes já lançados. Um Aviso existente também pode ser atualizado com novos CVEs , que são abordados pela atualização existente.

O Atualizações determinísticas por meio de repositórios versionados em 023 AL2 recurso do Amazon Linux 2023 significa que o repositório RPM de uma determinada versão AL2 023 contém um instantâneo dos metadados do repositório RPM a partir dessa versão. Isso inclui os metadados que descrevem as atualizações de segurança. O repositório RPM para uma determinada versão AL2 023 não é atualizado após o lançamento. Avisos de segurança novos ou atualizados não estarão visíveis ao examinar uma versão mais antiga dos repositórios AL2 023 RPM. Consulte a Listando os avisos

aplicáveis seção para saber como usar o gerenciador de `dnf` pacotes para examinar a versão do `latest` repositório ou uma versão AL2 023 específica.

Consultivo IDs

Cada Consultoria é referida por um`id`. Atualmente, é uma peculiaridade do Amazon Linux, onde o site do Amazon Linux Security Center listará um aviso como ALAS-2024-581, enquanto o gerenciador de `dnf` pacotes listará esse aviso como tendo o ID 023-2024-581. ALAS2 Quando Aplicando atualizações de segurança no local o ID do gerenciador de pacotes precisa ser usado para se referir a um aviso específico.

Para o Amazon Linux, cada versão principal do sistema operacional tem seu próprio namespace de Advisory. IDs Não se deve fazer suposições quanto ao formato do Amazon Linux Advisory. IDs Historicamente, a Amazon Linux Advisory IDs seguiu o padrão de`NAMESPACE-YEAR-NUMBER`. O intervalo completo de valores possíveis para não `NAMESPACE` está definido, mas inclui `ALASALASCORRETTO8`, `ALAS2023ALAS2`,`ALASPYTHON3.8`,, `ALASUNBOUND-1.17` e. `YEAR`Foi o ano em que a consultoria foi criada e `NUMBER` é um número inteiro exclusivo no namespace.

Embora o Advisory normalmente IDs seja sequencial e na ordem em que as atualizações sejam lançadas, há muitos motivos pelos quais esse não pode ser o caso, portanto, isso não deve ser assumido.

Trate o Advisory ID como uma string opaca que é exclusiva para cada versão principal do Amazon Linux.

No Amazon Linux 2, cada Extra estava em um repositório RPM separado, e os metadados do Advisory estão contidos somente no repositório ao qual são relevantes. Um aviso para um repositório não é aplicável a outro repositório. No site do Amazon Linux Security Center, há atualmente uma lista de recomendações para cada versão principal do Amazon Linux, e ela não está separada em listas por repositório.

Como o AL2 023 não usa o mecanismo Extras para empacotar versões alternativas de pacotes, atualmente existem apenas dois repositórios RPM, cada um com recomendações, o repositório e o `core` repositório. `livepatch` O `livepatch` repositório é paraAtualização do Kernel Live em 023 AL2.

Data de lançamento do comunicado e data de atualização do comunicado

A data de lançamento do aviso para Amazon Linux Advisories indica quando a atualização de segurança foi disponibilizada publicamente pela primeira vez no repositório RPM. Os avisos são

publicados no site do Amazon Linux Security Center imediatamente após as correções serem disponibilizadas para instalação por meio do repositório RPM.

A data de atualização do comunicado indica quando novas informações foram adicionadas a um comunicado após sua publicação anterior.

Não deve haver nenhuma suposição entre o número da versão AL2 023 (por exemplo, 2023.6.20241031) e a data de lançamento do comunicado dos avisos publicados junto com esse lançamento.

Tipos de consultoria

Os metadados do repositório RPM oferecem suporte a recomendações de diferentes tipos. Embora o Amazon Linux tenha emitido quase universalmente apenas avisos que são atualizações de segurança, isso não deve ser assumido. É possível que avisos para eventos como correções de bugs, aprimoramentos e novos pacotes possam ser emitidos, e o aviso seja marcado como contendo esse tipo de atualização.

Severidades consultivas

Cada Consultoria tem sua própria Gravidade, pois cada problema é avaliado separadamente. Várias CVEs podem ser tratadas em uma única Consultoria, e cada CVE pode ter uma avaliação diferente, mas a Consultoria em si tem uma Severidade. Pode haver várias recomendações referentes a uma única atualização de pacote, portanto, pode haver várias severidades para uma atualização de pacote específica (uma por recomendação).

Em ordem decrescente de severidade, o Amazon Linux usou Crítico, Importante, Moderado e Baixo para indicar a severidade de um aviso. Os Amazon Linux Advisories também podem não ter uma severidade, embora isso seja extremamente raro.

O Amazon Linux é uma das distribuições Linux baseadas em RPM que usa o termo Moderado, enquanto outras distribuições Linux baseadas em RPM usam o termo equivalente Médio. O gerenciador de pacotes Amazon Linux trata os dois termos como equivalentes, e repositórios de pacotes de terceiros podem usar o termo Médio.

As recomendações do Amazon Linux podem mudar a gravidade ao longo do tempo, à medida que se aprende mais sobre as questões relevantes abordadas na consultoria.

A severidade de um aviso normalmente rastreia a pontuação CVSS mais alta avaliada pelo Amazon Linux para aqueles CVEs referenciados pelo aviso. Pode haver casos em que esse não seja o caso. Um exemplo seria quando há um problema resolvido para o qual não há um CVE atribuído.

Consulte as perguntas frequentes do ALAS para obter mais informações sobre como o Amazon Linux usa as classificações de severidade consultivas.

Avisos e pacotes

Pode haver muitos avisos para um único pacote, e nem todos os pacotes terão um aviso publicado para eles. Uma versão específica do pacote pode ser referenciada em vários avisos, cada um com sua própria severidade e. CVEs

É possível que vários avisos para a mesma atualização de pacote sejam emitidos simultaneamente em uma nova versão AL2 023 ou em rápida sucessão.

Como outras distribuições Linux, pode haver um ou vários pacotes binários diferentes criados a partir do mesmo pacote fonte. Por exemplo, o ALAS-2024-698 é um aviso listado na seção AL2 023 do site do Amazon Linux Security Center como aplicável ao pacote. `mariadb105` Esse é o nome do pacote fonte, e o próprio Aviso se refere aos pacotes binários junto com o pacote fonte. Nesse caso, mais de uma dúzia de pacotes binários são criados a partir de um único pacote `mariadb105` fonte. Embora seja extremamente comum haver um pacote binário com o mesmo nome do pacote fonte, isso não é universal.

Embora o Amazon Linux Advisories normalmente liste todos os pacotes binários criados a partir do pacote fonte atualizado, isso não deve ser assumido. O gerenciador de pacotes e o formato de metadados do repositório RPM permitem recomendações que listam um subconjunto dos pacotes binários atualizados.

Um aviso específico também pode ser aplicado somente a uma arquitetura de CPU específica. Pode haver pacotes que não foram criados para todas as arquiteturas ou problemas que não afetam todas as arquiteturas. No caso em que um pacote está disponível em todas as arquiteturas, mas um problema se aplica somente a uma, o Amazon Linux normalmente não emitiu um aviso referenciando apenas a arquitetura afetada, embora isso não deva ser assumido.

Devido à natureza das dependências do pacote, é comum que um Aviso faça referência a um pacote, mas a instalação dessa atualização exigirá outras atualizações de pacotes, incluindo pacotes que não estão listados no Aviso. O gerenciador de `dnf` pacotes cuidará da instalação das dependências necessárias.

Avisos e CVEs

Uma Consultoria pode abordar zero ou mais CVEs, e pode haver várias Consultorias referenciando a mesma CVE.

Um exemplo de quando uma Consultoria pode fazer referência a zero CVEs é quando um CVE ainda não foi (ou nunca) foi atribuído ao problema.

Um exemplo de onde várias recomendações podem fazer referência ao mesmo CVE quando (por exemplo) o CVE é aplicável a vários pacotes. Por exemplo, CVE-2024-21208 se aplica ao Corretto 8, 11, 17 e 21. Cada uma dessas versões do Corretto é um pacote separado no AL2 023, e há um Aviso para cada um desses pacotes: ALAS-2024-754 para o Corretto 8, ALAS-2024-753 para o Corretto 11, ALAS-2024-752 para o Corretto 17 e ALAS-2024-752 para o Corretto 21. Embora todos esses lançamentos do Corretto tenham a mesma lista de CVEs, isso não deve ser assumido.

Um CVE específico pode ser avaliado de forma diferente para pacotes diferentes. Por exemplo, se um CVE específico for referenciado em um Aviso com uma Gravidade Importante, é possível que outro Aviso seja emitido referenciando o mesmo CVE com uma Gravidade diferente.

Os metadados do repositório RPM permitem uma lista de referências para cada Consultoria. Embora o Amazon Linux normalmente só faça referências CVEs, o formato de metadados permite outros tipos de referência.

Os metadados do repositório de pacotes RPM só se referirão CVEs com uma correção disponível. A seção Explore do site do Amazon Linux Security Center contém informações sobre o CVEs que o Amazon Linux avaliou. Essa avaliação pode resultar em uma pontuação básica, severidade e status do CVSS para várias versões e pacotes do Amazon Linux. O status de um CVE para uma versão ou pacote específico do Amazon Linux pode ser Não afetado, Correção pendente ou Nenhuma correção planejada. O status e a avaliação do CVEs podem mudar várias vezes e de qualquer forma antes da emissão de um Aviso. Isso inclui a reavaliação da aplicabilidade de um CVE ao Amazon Linux.

A lista de pessoas CVEs referenciadas por um Consultório pode mudar após a publicação inicial desse Aviso.

Texto consultivo

Um Aviso também conterá um texto descrevendo o problema ou problemas que foram o motivo da criação do Aviso. É comum que esse texto seja o texto CVE não modificado. Esse texto pode se referir aos números de versão upstream em que uma correção está disponível e que são diferentes

da versão do pacote à qual o Amazon Linux aplicou uma correção. É comum que o Amazon Linux faça backport de correções de versões upstream mais recentes. Caso o texto do Aviso mencione uma versão upstream diferente da versão enviada em uma versão do Amazon Linux, as versões do pacote Amazon Linux no Aviso serão precisas para o Amazon Linux.

É possível que o texto consultivo nos metadados do repositório RPM seja um texto reservado simplesmente referindo-se ao site do Amazon Linux Security Center para obter detalhes.

Avisos sobre o Kernel Live Patch

Os avisos para patches ativos são exclusivos, pois se referem a um pacote diferente (o kernel Linux) do pacote contra o qual o Aviso se refere (por exemplo). `kernel-livepatch-6.1.15-28.43`

Um aviso para um Kernel Live Patch fará referência aos problemas (como CVEs) que o pacote específico do Live Patch pode resolver para a versão específica do kernel à qual o pacote de patch ativo se aplica.

Cada patch ativo é para uma versão específica do kernel. Para aplicar um patch ativo a um CVE, o pacote de patch ativo correto para sua versão do kernel precisa ser instalado e o patch ativo aplicado.

Por exemplo, o CVE-2023-6111 pode ser corrigido ao vivo para AL2 as versões 023 do kernel, e. `6.1.56-82.125` `6.1.59-84.139` `6.1.61-85.141` Uma nova versão do kernel com uma correção para esse CVE também foi lançada e tem um aviso separado. Para que o CVE-2023-6111 seja endereçado no AL2 023, uma versão do kernel igual ou posterior à especificada pelo ALAS2023-2023-461 precisa estar em execução ou uma das versões do kernel com um patch ativo para esse CVE precisa estar em execução com o livepatch aplicável aplicado.

Quando há novos patches ativos disponíveis para uma versão específica do kernel que já tem um patch ativo disponível, uma nova versão do `kernel-livepatch-KERNEL_VERSION` pacote é lançada. Por exemplo, o ALASLIVEPATCH-2023-003Aviso foi emitido com o `kernel-livepatch-6.1.15-28.43-1.0-1.amzn2023` pacote que continha patches ativos para o `6.1.15-28.43` kernel, abrangendo três CVEs. Posteriormente, o ALASLIVEPATCH-2023-009Aviso foi lançado com o `kernel-livepatch-6.1.15-28.43-1.0-2.amzn2023` pacote; uma atualização do pacote de patch ativo anterior para o `6.1.15-28.43` kernel contendo patches ativos para outros três CVEs. Também houve outros problemas de recomendações de patches ativos para outras versões do kernel, com pacotes contendo patches ativos para essas versões específicas do kernel.

Para obter mais informações sobre o kernel live patching, consulte. Atualização do Kernel Live em 023 AL2

Para qualquer pessoa que desenvolva ferramentas relacionadas a alertas de segurança, também é recomendável consultar a Esquema XML para recomendações e `updateinfo.xml` seção para obter mais informações.

Esquema XML para recomendações e **`updateinfo.xml`**

O `updateinfo.xml` arquivo faz parte do formato do repositório de pacotes. São os metadados que o gerenciador de `dnf` pacotes analisa para implementar funcionalidades como e. Listando os avisos aplicáveis Aplicando atualizações de segurança no local

Recomendamos que a API do gerenciador de `dnf` pacotes seja usada em vez de escrever código personalizado para analisar os formatos de metadados do repositório. A versão do `dnf` in AL2 023 pode analisar os formatos AL2 023 e do AL2 repositório e, portanto, a API pode ser usada para examinar informações consultivas para qualquer versão do sistema operacional.

O projeto RPM Software Management documenta os formatos de metadados RPM no repositório rpm-metadata em. GitHub

Para aqueles que desenvolvem ferramentas para analisar diretamente os `updateinfo.xml` metadados, é altamente recomendável prestar muita atenção à documentação do rpm-metadata. A documentação aborda o que foi visto na natureza, o que inclui muitas exceções ao que você pode interpretar razoavelmente como uma regra para o formato de metadados.

Também há um conjunto crescente de exemplos reais de `updateinfo.xml` arquivos no repositório raw-historical-rpm-repository-examples em. GitHub

Caso algo não esteja claro na documentação, você pode abrir um problema no GitHub projeto para que possamos responder à pergunta e atualizar a documentação adequadamente. Como projetos de código aberto, pull requests atualizando a documentação também são bem-vindos.

Listando os avisos aplicáveis

O gerenciador de `dnf` pacotes tem acesso aos metadados que descrevem quais recomendações foram corrigidas em quais versões do pacote. Assim, ele pode listar quais recomendações são aplicáveis a uma instância ou imagem de contêiner.

> **Note**
>
> Ferramentas como a AWS Systems Managerpodem usar essa funcionalidade para mostrar quais atualizações são relevantes em uma frota, em vez de apenas em uma única instância.

Ao listar atualizações, você pode `dnf` instruir a examinar os metadados de uma versão AL2 023 específica ou os metadados da versão mais recente.

> **Note**
>
> Depois que uma versão AL2 023 é feita, ela é imutável. Assim, recomendações novas ou atualizadas sobre o Amazon Linux Security Center são adicionadas somente aos metadados das novas versões de 0.2.3. AL2

Agora, veremos exemplos de como ver quais recomendações se aplicam a cerca de AL2 023 imagens de contêiner. Todos esses comandos funcionam em ambientes não conteinerizados, como instâncias. EC2

Listing advisories in a specific version

Neste exemplo, veremos quais recomendações na versão 2023.1.20230628 são relevantes em uma imagem de contêiner da versão 2023.0.20230315.

> **Note**
>
> Este exemplo usa as versões 2023.0.20230315 e 2023.1.20230628, e essas não são a versão mais recente da 023. Consulte as notas de versão AL2 023 para ver as versões mais recentes, que contêm as AL2 atualizações de segurança mais recentes.

Neste exemplo, começaremos com uma imagem de contêiner para a versão 2023.0.20230315.

Primeiro, buscamos essa imagem do contêiner no registro do contêiner. O `.0` final indica a versão da imagem para uma versão específica; essa versão da imagem geralmente é zero.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

```
   2023.0.20230315.0: Pulling from amazonlinux/amazonlinux
b76f3b09316a: Pull complete
Digest: sha256:94e7183b0739140dbd5b639fb7600f0a2299cec5df8780c26d9cb409da5315a9
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.0.20230315.0
public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

Agora podemos gerar um shell dentro do contêiner, a partir do qual `dnf` solicitaremos uma lista de recomendações relevantes para os pacotes instalados no contêiner.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
   bash-5.2#
```

O `dnf updateinfo` comando agora é usado para exibir um resumo de quais recomendações na versão 2023.1.20230628 são relevantes para nossos pacotes instalados.

```
$ dnf updateinfo --releasever=2023.1.20230628
Amazon Linux 2023 repository                    42 MB/s |  15 MB     00:00
Last metadata expiration check: 0:00:02 ago on Mon Jul 22 20:24:24 2024.
Updates Information Summary: available
    8 Security notice(s)
        1 Important Security notice(s)
        5 Medium Security notice(s)
        2 Low Security notice(s)
```

Para obter uma lista dos avisos, a `--list` opção pode ser dada a. `dnf updateinfo`

```
$ dnf updateinfo --releasever=2023.1.20230628 --list
Last metadata expiration check: 0:01:22 ago on Mon Jul 22 20:24:24 2024.
ALAS2023-2023-193 Medium/Sec.    curl-minimal-8.0.1-1.amzn2023.x86_64
ALAS2023-2023-225 Medium/Sec.    glib2-2.74.7-688.amzn2023.0.1.x86_64
ALAS2023-2023-195 Low/Sec.       libcap-2.48-2.amzn2023.0.3.x86_64
ALAS2023-2023-193 Medium/Sec.    libcurl-minimal-8.0.1-1.amzn2023.x86_64
ALAS2023-2023-145 Low/Sec.       libgcc-11.3.1-4.amzn2023.0.3.x86_64
ALAS2023-2023-145 Low/Sec.       libgomp-11.3.1-4.amzn2023.0.3.x86_64
ALAS2023-2023-145 Low/Sec.       libstdc++-11.3.1-4.amzn2023.0.3.x86_64
ALAS2023-2023-163 Medium/Sec.    libxml2-2.10.4-1.amzn2023.0.1.x86_64
ALAS2023-2023-220 Important/Sec. ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch
ALAS2023-2023-220 Important/Sec. ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64
ALAS2023-2023-181 Medium/Sec.    openssl-libs-1:3.0.8-1.amzn2023.0.2.x86_64
ALAS2023-2023-222 Medium/Sec.    openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64
```

Listing advisories in the latest version

Neste exemplo, veremos quais atualizações estão disponíveis na versão AL2 023 se lançarmos um contêiner da `latest` versão 2023.4.20240319. No momento em que este artigo foi escrito, a `latest` versão era 2023.5.20240708, portanto, as atualizações listadas neste exemplo serão a partir dessa versão.

> **Note**
>
> Este exemplo usa as versões 2023.4.20240319 e 2023.5.20240708, sendo a última a versão mais recente no momento em que este artigo foi escrito. Para obter mais informações sobre as versões mais recentes, consulte as notas de versão AL2 023.

Neste exemplo, começaremos com uma imagem de contêiner para a versão 2023.4.20240319.

Primeiro, buscamos essa imagem do contêiner no registro do contêiner. O `.1` final indica a versão da imagem para uma versão específica. Embora a versão da imagem geralmente seja zero, este exemplo usa uma versão em que a versão da imagem é uma.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
   2023.4.20240319.1: Pulling from amazonlinux/amazonlinux
6de065fda9a2: Pull complete
Digest: sha256:b4838c4cc9211d966b6ea158dacc9eda7433a16ba94436508c2d9f01f7658b4e
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.4.20240319.1
public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

Agora podemos gerar uma concha dentro do contêiner, a partir da qual verificaremos se há atualizações.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
   bash-5.2#
```

O `dnf updateinfo` comando agora é usado para exibir um resumo de quais recomendações na versão mais recente são relevantes para nossos pacotes instalados. No momento em que este artigo foi escrito, 2023.1.20230628 era a versão mais recente.

```
$ dnf --releasever=latest updateinfo
Amazon Linux 2023 repository                    76 MB/s |  25 MB     00:00
```

```
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 20:59:54 2024.
Updates Information Summary: available
    9 Security notice(s)
        4 Important Security notice(s)
        4 Medium Security notice(s)
        1 Low Security notice(s)
```

Para obter uma lista dos avisos, a `--list` opção pode ser dada a. `dnf updateinfo`

```
$ dnf updateinfo --releasever=latest --list
Last metadata expiration check: 0:00:58 ago on Mon Jul 22 20:59:54 2024.
ALAS2023-2024-581 Low/Sec.       curl-minimal-8.5.0-1.amzn2023.0.3.x86_64
ALAS2023-2024-596 Medium/Sec.    curl-minimal-8.5.0-1.amzn2023.0.4.x86_64
ALAS2023-2024-576 Important/Sec. expat-2.5.0-1.amzn2023.0.4.x86_64
ALAS2023-2024-589 Important/Sec. glibc-2.34-52.amzn2023.0.10.x86_64
ALAS2023-2024-589 Important/Sec. glibc-common-2.34-52.amzn2023.0.10.x86_64
ALAS2023-2024-589 Important/Sec. glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64
ALAS2023-2024-586 Medium/Sec.    krb5-libs-1.21-3.amzn2023.0.4.x86_64
ALAS2023-2024-581 Low/Sec.       libcurl-minimal-8.5.0-1.amzn2023.0.3.x86_64
ALAS2023-2024-596 Medium/Sec.    libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64
ALAS2023-2024-592 Important/Sec. libnghttp2-1.59.0-3.amzn2023.0.1.x86_64
ALAS2023-2024-640 Medium/Sec.    openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64
ALAS2023-2024-605 Medium/Sec.    python3-3.9.16-1.amzn2023.0.7.x86_64
ALAS2023-2024-616 Important/Sec. python3-3.9.16-1.amzn2023.0.8.x86_64
ALAS2023-2024-605 Medium/Sec.    python3-libs-3.9.16-1.amzn2023.0.7.x86_64
ALAS2023-2024-616 Important/Sec. python3-libs-3.9.16-1.amzn2023.0.8.x86_64
```

Aplicando atualizações de segurança no local

Para obter uma visão geral da aplicação de atualizações, consulteAplicando atualizações de segurança usando DNF e versões do repositório. A `--security` opção de restringir `dnf upgrade` as atualizações de pacotes somente àqueles que têm um Aviso. O restante desta seção abordará como instalar somente atualizações de segurança específicas.

> **Note**
>
> É recomendável aplicar todas as atualizações disponíveis em uma nova versão AL2 023. Escolher apenas atualizações de segurança ou apenas atualizações específicas deve ser a exceção e não a regra.

Aplicando as atualizações mencionadas em um Aviso

Os identificadores de recomendação na primeira coluna da saída de `dnf upgradeinfo` podem ser usados para aplicar atualizações para os pacotes mencionados na recomendação. O gerenciador de `dnf` pacotes pode ser instruído a atualizar os pacotes no comunicado para as versões mais recentes disponíveis ou somente até as versões mencionadas no comunicado. Se as atualizações já estiverem instaladas, o comando update é autônomo.

Para aplicar as atualizações aos pacotes afetados somente até a versão mencionada no comunicado, use o `dnf upgrade-minimal` comando enquanto usa a `--advisory` opção para especificar o aviso. O exemplo a seguir está sendo executado `dnf upgrade-minimal` em um contêiner AL2 023 versão 2023.0.20230315.

```
$ dnf upgrade-minimal -y --releasever=2023.1.20230628 --advisory ALAS2023-2023-193
Amazon Linux 2023 repository                    46 MB/s |  15 MB     00:00
Last metadata expiration check: 0:00:03 ago on Mon Jul 22 20:36:13 2024.
Dependencies resolved.
================================================================================
 Package              Arch        Version                Repository        Size
================================================================================
Upgrading:
 curl-minimal         x86_64      8.0.1-1.amzn2023       amazonlinux      150 k
 libcurl-minimal      x86_64      8.0.1-1.amzn2023       amazonlinux      249 k

Transaction Summary
================================================================================
Upgrade  2 Packages

Total download size: 399 k
Downloading Packages:
(1/2): curl-minimal-8.0.1-1.amzn2023.x86_64.rpm 2.7 MB/s | 150 kB     00:00
(2/2): libcurl-minimal-8.0.1-1.amzn2023.x86_64. 3.8 MB/s | 249 kB     00:00
--------------------------------------------------------------------------------
Total                                           2.5 MB/s | 399 kB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Upgrading        : libcurl-minimal-8.0.1-1.amzn2023.x86_64                1/4
  Upgrading        : curl-minimal-8.0.1-1.amzn2023.x86_64                   2/4
```

```
  Cleanup          : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64              3/4
  Cleanup          : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64           4/4
  Running scriptlet: libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64           4/4
  Verifying        : libcurl-minimal-8.0.1-1.amzn2023.x86_64                1/4
  Verifying        : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64           2/4
  Verifying        : curl-minimal-8.0.1-1.amzn2023.x86_64                   3/4
  Verifying        : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64              4/4

Upgraded:
  curl-minimal-8.0.1-1.amzn2023.x86_64  libcurl-minimal-8.0.1-1.amzn2023.x86_64

Complete!
```

As mesmas versões do pacote são atualizadas mesmo se forem `--releasever=latest` usadas, pois a solicitação é `dnf` para fazer a atualização mínima necessária para abordar o aviso.

Usar o `dnf upgrade` comando regular com a `--advisory` opção atualizará os pacotes relevantes mencionados no comunicado para a versão mais recente disponível, que pode ser mais recente do que a versão mencionada no comunicado.

> **Note**
>
> A menos que o `system-release` pacote seja atualizado, a versão dos repositórios AL2 023 bloqueados não muda. `dnf`

> **Warning**
>
> Ao instalar atualizações de uma versão diferente do AL2 023 sem alterar a versão do repositório bloqueado, deve-se tomar cuidado com qualquer operação de mutação `dnf` subsequente. `dnf` Por exemplo, ao instalar ou atualizar pacotes, como as dependências do pacote podem ter sido alteradas na versão mais recente, a versão mais antiga em que você permanece pode não ser capaz de satisfazer essas novas dependências.

O exemplo a seguir é executado em um contêiner AL2 023 versão 2023.0.20230315 referente à versão mais recente de AL2 023, cuja data de gravação foi 2023.5.20240708. Observe que a versão da `curl` atualização é mais recente do que a versão `update-minimal` atualizada, mas essa versão mais recente traz novas dependências.

```
$ dnf upgrade -y --releasever=latest --advisory ALAS2023-2023-193
Amazon Linux 2023 repository                    80 MB/s |  25 MB     00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 20:48:38 2024.
Dependencies resolved.
================================================================================
 Package                  Arch     Version                 Repository     Size
================================================================================
Upgrading:
 curl-minimal             x86_64   8.5.0-1.amzn2023.0.4    amazonlinux   160 k
 libcurl-minimal          x86_64   8.5.0-1.amzn2023.0.4    amazonlinux   275 k
 libnghttp2               x86_64   1.59.0-3.amzn2023.0.1   amazonlinux    79 k
Installing dependencies:
 libpsl                   x86_64   0.21.1-3.amzn2023.0.2   amazonlinux    61 k
 publicsuffix-list-dafsa  noarch   20240212-61.amzn2023    amazonlinux    59 k

Transaction Summary
================================================================================
Install  2 Packages
Upgrade  3 Packages

Total download size: 634 k
Downloading Packages:
(1/5): publicsuffix-list-dafsa-20240212-61.amzn 1.1 MB/s |  59 kB     00:00
(2/5): curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 2.6 MB/s | 160 kB     00:00
(3/5): libpsl-0.21.1-3.amzn2023.0.2.x86_64.rpm  949 kB/s |  61 kB     00:00
(4/5): libnghttp2-1.59.0-3.amzn2023.0.1.x86_64. 3.7 MB/s |  79 kB     00:00
(5/5): libcurl-minimal-8.5.0-1.amzn2023.0.4.x86 6.7 MB/s | 275 kB     00:00
--------------------------------------------------------------------------------
Total                                           3.5 MB/s | 634 kB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Upgrading        : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64                1/8
  Installing       : publicsuffix-list-dafsa-20240212-61.amzn2023.noarch    2/8
  Installing       : libpsl-0.21.1-3.amzn2023.0.2.x86_64                    3/8
  Upgrading        : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64            4/8
  Upgrading        : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64               5/8
  Cleanup          : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64              6/8
  Cleanup          : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64           7/8
  Cleanup          : libnghttp2-1.51.0-1.amzn2023.x86_64                    8/8
```

```
  Running scriptlet: libnghttp2-1.51.0-1.amzn2023.x86_64                    8/8
  Verifying        : libpsl-0.21.1-3.amzn2023.0.2.x86_64                    1/8
  Verifying        : publicsuffix-list-dafsa-20240212-61.amzn2023.noarch    2/8
  Verifying        : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64               3/8
  Verifying        : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64              4/8
  Verifying        : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64            5/8
  Verifying        : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64           6/8
  Verifying        : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64                7/8
  Verifying        : libnghttp2-1.51.0-1.amzn2023.x86_64                    8/8

Upgraded:
  curl-minimal-8.5.0-1.amzn2023.0.4.x86_64
  libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64
  libnghttp2-1.59.0-3.amzn2023.0.1.x86_64
Installed:
  libpsl-0.21.1-3.amzn2023.0.2.x86_64
  publicsuffix-list-dafsa-20240212-61.amzn2023.noarch

Complete!
```

SELinux Modos de configuração para AL2 023

Por padrão, o Security Enhanced Linux (SELinux) está `enabled` configurado para o `permissive` modo AL2 023. No modo permissivo, as negações de permissão são registradas, mas não aplicadas. SELinux é uma coleção de recursos e utilitários do kernel para fornecer uma arquitetura de controle de acesso (MAC) forte, flexível e obrigatória aos principais subsistemas do kernel.

SELinux fornece um mecanismo aprimorado para impor a separação de informações com base nos requisitos de confidencialidade e integridade. Essa separação de informações reduz as ameaças de adulteração e desvio dos mecanismos de segurança do aplicativo. Também limita os danos que podem ser causados por aplicativos maliciosos ou defeituosos.

SELinux inclui um conjunto de exemplos de arquivos de configuração de políticas de segurança projetados para atender às metas diárias de segurança.

Para obter mais informações sobre SELinux recursos e funcionalidades, consulte SELinux Notebook and Policy Languages".

Tópicos

- SELinux Status e modos padrão para AL2 0.23
- Mudar para o modo enforcing

- Opção para desativar SELinux para AL2 023

SELinux Status e modos padrão para AL2 0.23

Para AL2 023, SELinux por padrão é `enabled` e definido como `permissive` modo. No modo `permissive`, as negações de permissão são registradas, mas não aplicadas.

Os **`sestatus`** comandos **`getenforce`** or informam o SELinux status, a política e o modo atuais.

Com o status padrão definido como `enabled` e `permissive`, o comando **`getenforce`** retorna `permissive`.

O **`sestatus`** comando retorna o SELinux status e a SELinux política atual, conforme mostrado no exemplo a seguir:

```
$ sestatus
SELinux status:                 enabled
  SELinuxfs mount:                /sys/fs/selinux
  SELinux root directory:         /etc/selinux
  Loaded policy name:             targeted
  Current mode:                   permissive
  Mode from config file:          permissive
  Policy MLS status:              enabled
  Policy deny_unknown status:     allowed
  Memory protection checking:     actual (secure)
  Max kernel policy version:      33
```

Quando você executa SELinux no `permissive` modo, os usuários podem rotular arquivos incorretamente. Quando você SELinux executa o `disabled` status, os arquivos não são rotulados. Arquivos incorretos ou não identificados podem causar problemas quando você muda para o modo `enforcing`.

SELinux renomeia automaticamente os arquivos para evitar esse problema. SELinux evita problemas de etiquetagem com a nova etiquetagem automática quando você altera o status para. `enabled`

Mudar para o modo **`enforcing`**

Quando você corre SELinux no `enforcing` modo, o SELinux utilitário é `enforcing` a política configurada. SELinux controla os recursos de aplicativos selecionados ao permitir ou negar o acesso com base nas regras da política.

Para encontrar o atual SELinux modo, execute o `getenforce` comando.

```
getenforce
Permissive
```

Edite o arquivo de configuração para ativar o modo **enforcing**

Para alterar o modo para`enforcing`, use as etapas a seguir.

1. Edite o arquivo `enforcing` para mudar para o modo `/etc/selinux/config`. A `SELINUX` configuração deve ser semelhante ao exemplo a seguir.

   ```
   SELINUX=enforcing
   ```

2. Reinicie o sistema para concluir a mudança para o modo `enforcing`.

   ```
   $ sudo reboot
   ```

Na próxima bota, SELinux renomeia todos os arquivos e diretórios no sistema. SELinux também adiciona o SELinux contexto para arquivos e diretórios que foram criados quando SELinux foi`disabled`.

Depois de mudar para o `enforcing` modo, SELinux pode negar algumas ações por causa de ações incorretas ou ausentes SELinux regras de política. Você pode ver as ações que SELinux nega com o seguinte comando.

```
$ sudo ausearch -m AVC,USER_AVC,SELINUX_ERR,USER_SELINUX_ERR -ts recent
```

Use cloud-init para ativar o **enforcing** modo

Como alternativa, ao iniciar sua instância, transmita o seguinte `cloud-config` como dados do usuário para ativar o modo `enforcing`.

```
#cloud-config
selinux:
  mode: enforcing
```

Por padrão, essa configuração faz com que a instância seja reinicializada. Para maior estabilidade, recomendamos reinicializar sua instância. Entretanto, se você preferir, poderá pular a reinicialização fornecendo o `cloud-config` a seguir.

```
#cloud-config
selinux:
  mode: enforcing
  selinux_no_reboot: 1
```

Opção para desativar SELinux para AL2 023

Quando você desativa SELinux, SELinux a política não é carregada nem aplicada e as mensagens do Access Vector Cache (AVC) não são registradas. Você perde todos os benefícios da corrida SELinux.

Em vez de desativar SELinux, recomendamos usar o `permissive` modo. Custa apenas um pouco mais executar no `permissive` modo do que desabilitar SELinux completamente. A transição de um `permissive` `enforcing` modo para outro requer muito menos ajuste de configuração do que a transição de volta ao modo após a desativação `enforcing` SELinux. Você pode rotular arquivos e o sistema pode rastrear e registrar ações que a política ativa possa ter negado.

Alteração SELinux para o **`permissive`** modo

Quando você corre SELinux no `permissive` modo, SELinux a política não é aplicada. No `permissive` modo, SELinux registra as mensagens do AVC, mas não nega as operações. Você pode usar essas mensagens do AVC para solução de problemas, depuração e SELinux melhorias na política.

Para mudar SELinux para o modo permissivo, use as etapas a seguir.

1. Edite o arquivo `permissive` para mudar para o modo `/etc/selinux/config`. O `SELINUX` valor deve ser semelhante ao exemplo a seguir.

   ```
   SELINUX=permissive
   ```

2. Reinicie o sistema para concluir a mudança para o modo `permissive`.

   ```
   sudo reboot
   ```

Disable (Desabilitar) SELinux

Quando você desativa SELinux, SELinux a política não é carregada nem aplicada, e as mensagens AVC não são registradas. Você perde todos os benefícios da corrida SELinux.

Para desativar SELinux, use as etapas a seguir.

1. Certifique-se de que o `grubby` pacote esteja instalado.

```
rpm -q grubby
grubby-version
```

2. Configure seu bootloader para adicionar `selinux=0` à linha de comando do kernel.

```
sudo grubby --update-kernel ALL --args selinux=0
```

3. Reinicie o sistema.

```
sudo reboot
```

4. Execute o `getenforce` comando para confirmar que SELinux é`Disabled`.

```
$ getenforce
Disabled
```

Para obter mais informações sobre SELinux, veja o SELinux Caderno e SELinux configuração.

Ative o modo FIPS em 023 AL2

Esta seção explica como habilitar os Padrões Federais de Processamento de Informações (FIPS) em AL2 023. Para obter mais informações sobre FIPS, consulte:

- Federal Information Processing Standard (FIPS)
- Conformidade FAQs: Padrões federais de processamento de informações

> **Note**
>
> Esta seção documenta como habilitar FIPS No modo AL2 023, ele não cobre o status de certificação dos módulos criptográficos AL2 023.

Pré-requisitos

- Uma EC2 instância AL2 023 (AL2023.2 ou superior) existente da Amazon com acesso à Internet para baixar os pacotes necessários. Para obter mais informações sobre o lançamento de uma EC2 instância AL2 023 da Amazon, consulteLançamento do AL2 023 usando o console da Amazon EC2 .
- Você deve se conectar à sua EC2 instância da Amazon usando SSH ou AWS Systems Manager. Para obter mais informações, consulte Conexão com AL2 203 instâncias.

> **Important**
>
> ED25519 As chaves de usuário SSH não são suportadas no modo FIPS. Se você iniciou sua EC2 instância Amazon usando um par de chaves ED25519 SSH, deverá gerar novas chaves usando outro algoritmo (como RSA) ou poderá perder o acesso à sua instância após ativar o modo FIPS. Para obter mais informações, consulte Criar pares de chaves no Guia EC2 do usuário da Amazon.

Habilitar o modo FIPS

1. Conecte-se à sua instância AL2 023 usando SSH ou. AWS Systems Manager
2. Verifique se o sistema está atualizado. Para obter mais informações, consulte Gerencie atualizações de pacotes e sistemas operacionais em AL2 023.
3. Certifique-se de que os `crypto-policies` utilitários estejam instalados e. up-to-date

   ```
   sudo dnf -y install crypto-policies crypto-policies-scripts
   ```

4. Ativar modo FIPS executando o seguinte comando. Isso habilitará o modo FIPS em todo o sistema para os módulos listados nas perguntas frequentes do 023 AL2

   ```
   sudo fips-mode-setup --enable
   ```

5. Execute a instância usando o seguinte comando.

```
sudo reboot
```

6. Para verificar se o modo do FIPS está habilitado, reconecte-se à sua instância e execute o comando a seguir.

```
sudo fips-mode-setup --check
```

O exemplo de saída a seguir mostra que o modo do FIPS está habilitado:

```
FIPS mode is enabled.
```

Ativar o modo FIPS em um contêiner AL2 023

Esta seção explica como ativar os Padrões Federais de Processamento de Informações (FIPS) em um contêiner AL2 023. Para obter mais informações sobre FIPS, consulte:

- Federal Information Processing Standard (FIPS)
- Conformidade FAQs: Padrões federais de processamento de informações

> **Note**
>
> Esta seção documenta como habilitar FIPS modo em um contêiner AL2 023. Ele não cobre o status de certificação de AL2 023 módulos criptográficos.

Pré-requisitos

- Uma EC2 instância AL2 023 (AL2023.2 ou superior) existente da Amazon com acesso à Internet para baixar os pacotes necessários. Para obter mais informações sobre o lançamento de uma EC2 instância AL2 023 da Amazon, consulteLançamento do AL2 023 usando o console da Amazon EC2 .
- Você deve se conectar à sua EC2 instância Amazon usando SSH ou AWS Systems Manager. Para obter mais informações, consulte Conexão com AL2 203 instâncias.

> **Important**
>
> O `fips-mode-setup` comando não funcionará corretamente de dentro do contêiner. Leia as etapas abaixo para configurar corretamente o modo FIPS em um contêiner AL2 023.

Ativar o modo FIPS em um contêiner AL2 023

1. O modo FIPS deve primeiro ser ativado no host do contêiner AL2 023. Siga as instruções em Ative o modo FIPS em 023 AL2 para ativar o modo FIPS no Host.
2. Conecte-se à sua instância de host AL2 de contêiner 023 usando SSH ou. AWS Systems Manager
3. O modo FIPS será ativado automaticamente em um contêiner AL2 023 se o host AL2 023 estiver no modo FIPS e puder ser `/proc/sys/crypto/fips_enabled` acessado de dentro do contêiner. Se o conteúdo de `/proc/sys/crypto/fips_enabled` `0` for, o FIPS não está ativado e um valor de 1 indica que o modo FIPS está ativado.

   Você pode verificar se o FIPS está ativado executando o seguinte comando no host AL2 023 e no contêiner:

   ```
   cat /proc/sys/crypto/fips_enabled
   ```

4. Em seguida, ative as crypto-policies do FIPS dentro do contêiner. Há várias maneiras de fazer isso, descritas nas opções abaixo. Use a opção que funciona melhor para seu ambiente.

   a. Ative as crypto-policies do FIPS manualmente dentro do contêiner usando o comando: `update-crypto-policies`

   ```
   # Run these commands inside the container
   dnf install -y crypto-policies-scripts
   update-crypto-policies --set FIPS
   ```

   b. Crie `bind` montagens dentro do contêiner AL2 023 (isso é semelhante ao que `podman` funciona em outras distribuições):

   ```
   # Run these commands inside the container
   mount --bind /usr/share/crypto-policies/back-ends/FIPS /etc/crypto-policies/back-ends
   echo "FIPS" > /usr/share/crypto-policies/default-fips-config
   ```

```
mount --bind /usr/share/crypto-policies/default-fips-config /etc/crypto-policies/
config
```

c. Também é possível criar uma montagem de associação para que o contêiner AL2 023 corresponda às crypto-policies do AL2 host 023. O seguinte é fornecido apenas como exemplo. Essa configuração pode causar problemas se houver diferenças incompatíveis nas crypto-policies e nas versões do pacote entre o contêiner e o host:

```
sudo docker pull amazonlinux:2023
sudo docker run --mount type=bind,readonly,src=/etc/crypto-policies,dst=/etc/
crypto-policies -it amazonlinux:2023
```

5. Depois de executar as etapas acima, você pode verificar novamente se o FIPS está ativado no contêiner com os seguintes comandos:

```
$ cat /etc/crypto-policies/config
FIPS

$ cat /proc/sys/crypto/fips_enabled
1
```

Troque provedores OpenSSL FIPS em 023 AL2

Esta seção explica como alternar entre os provedores `latest` FIPS e `certified` OpenSSL em 023. AL2

Para obter mais informações sobre FIPS, consulte:

- Federal Information Processing Standard (FIPS)
- Conformidade FAQs: Padrões federais de processamento de informações
- Política do FedRAMP para seleção e uso de módulos criptográficos

> **Important**
>
> Na AL2 versão 023.7 e superior, o provedor FIPS padrão do OpenSSL é o `openssl-fips-provider-latest` pacote, que recebe correções de bugs e atualizações de segurança regulares.

As instruções abaixo são apenas para clientes que desejam fixar na `openssl-fips-provider-certified` embalagem. Essa versão do provedor FIPS corresponderá à soma de verificação do certificado NIST e pode não ter as atualizações mais recentes. Consulte as perguntas frequentes do AL2 023 para obter mais informações sobre módulos certificados FIPS e versões de pacotes.

Pré-requisitos

- Uma EC2 instância AL2 023 (AL2023.7 ou superior) existente da Amazon com acesso à Internet para baixar os pacotes necessários. Para obter mais informações sobre o lançamento de uma EC2 instância AL2 023 da Amazon, consulteLançamento do AL2 023 usando o console da Amazon EC2 .
- Você deve se conectar à sua EC2 instância Amazon usando SSH ou AWS Systems Manager. Para obter mais informações, consulte Conexão com AL2 203 instâncias.
- Para ativar o modo FIPS em AL2 023, siga as instruções em. Ative o modo FIPS em 023 AL2

Alternar entre **`openssl-fips-provider-latest`** e **`openssl-fips-provider-certified`**

1. Use `dnf` para alternar o provedor FIPS do OpenSSL:

```
sudo dnf -y swap openssl-fips-provider-latest openssl-fips-provider-certified
```

2. Verifique se você está usando o provedor certificado de FIPS OpenSSL. Com AL2 023 no modo FIPS, execute o seguinte comando:

```
openssl list -providers
```

A seguinte saída deverá ser mostrada:

```
Providers:
  base
    name: OpenSSL Base Provider
    version: 3.2.2
    status: active
  default
    name: OpenSSL Default Provider
```

```
    version: 3.2.2
    status: active
  fips
    name: Amazon Linux 2023 - OpenSSL FIPS Provider
    version: 3.0.8-d694bfa693b76001
    status: active
```

AL2023 Endurecimento do kernel

O kernel Linux 6.1 em AL2 023 é configurado e construído com várias opções e recursos de fortalecimento.

Opções de fortalecimento do kernel (independente da arquitetura)

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_ACPI_CUSTOM_METHOD	n	n	N/D	N/D
CONFIG_BINFMT_MISC	m	m	m	m
CONFIG_BUG	y	y	y	y
CONFIG_BUG_ON_DATA_CORRUPTION	y	y	y	y
CONFIG_CFI_CLANG	N/D	N/D	N/D	N/D
CONFIG_CFI_PERMISSIVE	N/D	N/D	N/D	N/D

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_COMPAT	y	y	y	y
CONFIG_COMPAT_BRK	n	n	n	n
CONFIG_COMPAT_VDSO	N/D	n	N/D	n
CONFIG_DEBUG_CREDENTIALS	n	n	N/D	N/D
CONFIG_DEBUG_LIST	y	y	y	y
CONFIG_DEBUG_NOTIFIERS	n	n	n	n
CONFIG_DEBUG_SG	n	n	n	n
CONFIG_DEBUG_VIRTUAL	n	n	n	n
CONFIG_DEBUG_WX	n	n	n	n
CONFIG_DEFAULT_MMAP_MIN_ADDR	65536	65536	65536	65536
CONFIG_DEVKMEM	N/D	N/D	N/D	N/D

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_DEVMEM	n	n	n	n
CONFIG_EFI_DISABLE_PCI_DMA	n	n	n	n
CONFIG_FORTIFY_SOURCE	y	y	y	y
CONFIG_HARDENED_USERCOPY	y	y	y	y
CONFIG_HARDENED_USERCOPY_FALLBACK	N/D	N/D	N/D	N/D
CONFIG_HARDENED_USERCOPY_PAGESPAN	N/D	N/D	N/D	N/D
CONFIG_HIBERNATION	y	y	y	y
CONFIG_HW_RANDOM_TPM	N/D	N/D	N/D	N/D
CONFIG_INET_DIAG	m	m	m	m

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_INIT_ON_ALLOC_DEFAULT_ON	n	n	n	n
CONFIG_INIT_ON_FREE_DEFAULT_ON	n	n	n	n
CONFIG_INIT_STACK_ALL_ZERO	N/D	N/D	N/D	N/D
CONFIG_IOMMU_DEFAULT_DMA_STRICT	n	n	n	n
CONFIG_IOMMU_SUPPORT	y	y	y	y
CONFIG_IO_STRICT_DEVMEM	N/D	N/D	N/D	N/D
CONFIG_KEXEC	y	y	y	y
CONFIG_KFENCE	n	n	n	n

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_LDISC_AUTOLOAD	n	n	n	n
CONFIG_LEGACY_PTYS	n	n	n	n
CONFIG_LOCK_DOWN_KERNEL_FORCE_CONFIDENTIALITY	n	n	n	n
CONFIG_MODULES	y	y	y	y
CONFIG_MODULE_SIG	y	y	y	y
CONFIG_MODULE_SIG_ALL	y	y	y	y
CONFIG_MODULE_SIG_FORCE	n	n	n	n
CONFIG_MODULE_SIG_HASH	sha512	sha512	sha512	sha512
CONFIG_MODULE_SIG_KEY	certs/signing_key.pem	certs/signing_key.pem	certs/signing_key.pem	certs/signing_key.pem

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_MODULE_SIG_SHA512	y	y	y	y
CONFIG_PAGE_POISONING	n	n	n	n
CONFIG_PAGE_POISONING_NO_SANITY	N/D	N/D	N/D	N/D
CONFIG_PAGE_POISONING_ZERO	N/D	N/D	N/D	N/D
CONFIG_PANIC_ON_OOPS	y	y	y	y
CONFIG_PANIC_TIMEOUT	0	0	0	0
CONFIG_PROC_KCORE	y	y	y	y
CONFIG_RANDOMIZE_KSTACK_OFFSET_DEFAULT	n	n	n	n

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_RANDOM_TRUST_BOOTLOADER	y	y	N/D	N/D
CONFIG_RANDOM_TRUST_CPU	y	y	N/D	N/D
CONFIG_REFCOUNT_FULL	N/D	N/D	N/D	N/D
CONFIG_SCHED_CORE	N/D	y	N/D	y
CONFIG_SCHED_STACK_END_CHECK	y	y	y	y
CONFIG_SECCOMP	y	y	y	y
CONFIG_SECCOMP_FILTER	y	y	y	y
CONFIG_SECURITY	y	y	y	y
CONFIG_SECURITY_DMESG_RESTRICT	y	y	y	y

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_SECURITY_LANDLOCK	n	n	n	n
CONFIG_SECURITY_LOCKDOWN_LSM	y	y	y	y
CONFIG_SECURITY_LOCKDOWN_LSM_EARLY	y	y	y	y
CONFIG_SECURITY_SELINUX_BOOTPARAM	y	y	y	y
CONFIG_SECURITY_SELINUX_DEVELOP	y	y	y	y
CONFIG_SECURITY_SELINUX_DISABLE	n	n	N/D	N/D
CONFIG_SECURITY_WRITABLE_HOOKS	N/D	N/D	N/D	N/D

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_SECURITY_YAMA	y	y	y	y
CONFIG_SHUFFLE_PAGE_ALLOCATOR	y	y	y	y
CONFIG_SLAB_FREELIST_HARDENED	y	y	y	y
CONFIG_SLAB_FREELIST_RANDOM	y	y	y	y
CONFIG_SLUB_DEBUG	y	y	y	y
CONFIG_STACKPROTECTOR	y	y	y	y
CONFIG_STACKPROTECTOR_STRONG	y	y	y	y
CONFIG_STATIC_USERMODEHELPER	n	n	n	n

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_STRICT_DEVMEM	n	n	n	n
CONFIG_STRICT_KERNEL_RWX	y	y	y	y
CONFIG_STRICT_MODULE_RWX	y	y	y	y
CONFIG_SYN_COOKIES	y	y	y	y
CONFIG_VMAP_STACK	y	y	y	y
CONFIG_WERROR	n	n	n	n
CONFIG_ZERO_CALL_USED_REGS	n	n	n	n

Permitir que métodos ACPI sejam inseridos/substituídos em tempo de execução (CONFIG_ACPI_CUSTOM_METHOD)

O Amazon Linux desativa essa opção, pois permite que os usuários `root` gravem na memória arbitrária do kernel.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Formatos binários diversos (**`binfmt_misc`**)

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. No AL2 023, esse recurso é opcional e é construído como um módulo do kernel.

Ajuda do **`BUG()`**

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

`BUG()` se o kernel encontrar corrupção de dados ao verificar a validade das estruturas de memória do kernel

Algumas partes do kernel Linux verificarão a consistência interna das estruturas de dados e podem `BUG()` quando detectarem dados corrompidos.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

`COMPAT_BRK`

Com essa opção desativada (que é como o Amazon Linux configura o kernel), a configuração de `randomize_va_space sysctl` é retornada para 2, o que também permite a randomização de heap sobre o topo da base `mmap`, pilha e randomização da página VDSO.

Essa opção existe no kernel para fornecer compatibilidade com alguns binários `libc.so.5` antigos de 1996 e anteriores.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

`COMPAT_VDSO`

Essa opção de configuração é relevante para `x86-64` o `aarch64`. Ao definir isso como `n`, o kernel do Amazon Linux não torna um objeto compartilhado dinâmico (VDSO) virtual de 32 bits visível em um endereço previsível. A mais recente `glibc` conhecida por ser quebrada por essa opção sendo definida por `n` é `glibc` 2.3.3, de 2004.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

`CONFIG_DEBUG` fortalecimento fechado

As opções de configuração do kernel Linux gated by `CONFIG_DEBUG` são normalmente projetadas para uso em kernels criados para problemas de depuração, e coisas como desempenho não são uma prioridade. AL2023 ativa a opção de `CONFIG_DEBUG_LIST` endurecimento.

Desative o DMA para dispositivos PCI no stub EFI antes de configurar o IOMMU

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

Fortalecimento para copiar memória entre o kernel e o espaço do usuário

Quando o kernel precisa copiar a memória para ou do espaço do usuário, essa opção ativa algumas verificações que podem proteger contra algumas classes de problemas de estouro de pilha.

A opção `CONFIG_HARDENED_USERCOPY_FALLBACK` existia nos kernels 4.16 a 5.15 para ajudar os desenvolvedores do kernel a descobrir quaisquer entradas ausentes da lista de permissões por meio de uma `WARN()`. Como o AL2 023 vem com um kernel 6.1, essa opção não é mais relevante para o 023. AL2

A `CONFIG_HARDENED_USERCOPY_PAGESPAN` opção existia nos kernels principalmente como uma opção de depuração para desenvolvedores e não se aplica mais ao kernel 6.1 em 023. AL2

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Suporte de hibernação

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. Essa opção precisa ser ativada para oferecer suporte à capacidade de hibernar sua instância sob demanda e para oferecer suporte à capacidade de hibernar instâncias spot interrompidas.

Geração de números aleatórios

O kernel AL2 023 é configurado para garantir que a entropia adequada esteja disponível para uso interno. EC2

`CONFIG_INET_DIAG`

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. No AL2 023, esse recurso é opcional e é construído como um módulo do kernel.

Zere toda a memória do alocador de páginas e placas do kernel na alocação e desalocação

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. Essas opções são desativadas em AL2 023 devido ao possível impacto no desempenho da ativação dessa funcionalidade por padrão. O comportamento `CONFIG_INIT_ON_ALLOC_DEFAULT_ON` pode ser ativado adicionando `init_on_alloc=1` à linha de comando do kernel e o comportamento `CONFIG_INIT_ON_FREE_DEFAULT_ON` pode ser ativado adicionando `init_on_free=1`.

Inicialize todas as variáveis da pilha como zero (`CONFIG_INIT_STACK_ALL_ZERO`)

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. Esta opção requer GCC 12 ou superior, enquanto o AL2 023 vem com o GCC 11.

Assinatura do módulo Kernel

AL2023 assina e valida as assinaturas dos módulos do kernel. A opção `CONFIG_MODULE_SIG_FORCE`, que exigiria que os módulos tivessem uma assinatura válida, não está habilitada para preservar a compatibilidade dos usuários que criam módulos de terceiros. Para usuários que desejam garantir que todos os módulos do kernel sejam assinados, Módulo de segurança Linux (LSM) Lockdown pode ser configurado para impor isso.

`kexec`

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. Essa opção está ativada para que a funcionalidade `kdump` possa ser usada.

Suporte ao **IOMMU**

AL2023 ativa o suporte ao IOMMU. A opção `CONFIG_IOMMU_DEFAULT_DMA_STRICT` não está habilitada por padrão, mas essa funcionalidade pode ser configurada adicionando `iommu.passthrough=0 iommu.strict=1` à linha de comando do kernel.

kfence

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

Suporte de **pty** antigo

AL2023 usa o moderno PTY interface (`devpts`).

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Módulo de segurança Linux (LSM) Lockdown

AL2O 023 cria o `lockdown` LSM, que bloqueará automaticamente o kernel ao usar o Secure Boot.

A opção `CONFIG_LOCK_DOWN_KERNEL_FORCE_CONFIDENTIALITY` não está ativada. Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. Quando não estiver usando o Secure Boot, é possível habilitar o LSM de bloqueio e configurá-lo conforme desejado.

Envenenamento de páginas

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. Da mesma forma Zere toda a memória do alocador de páginas e placas do kernel na alocação e desalocação , isso está desativado no kernel AL2 023 devido ao possível impacto no desempenho.

Protetor de pilha

O kernel AL2 023 é construído com o recurso de proteção de pilha do GCC ativado com a `-fstack-protector-strong` opção.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

seccomp BPF API

A ferramenta seccomp o recurso de fortalecimento é usado por softwares, como `systemd` tempos de execução de contêineres, para fortalecer os aplicativos do espaço do usuário.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

tempo esgotado para **`panic()`**

O kernel AL2 023 é configurado com esse valor definido como`0`, o que significa que o kernel não será reinicializado após entrar em pânico. Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. Isso é configurável por meio de `sysctl`, `/proc/sys/kernel/panic` e na linha de comando do kernel.

Modelos de segurança

AL2O 023 é ativado SELinux no modo Permissivo por padrão. Para obter mais informações, consulte SELinux Modos de configuração para AL2 023.

Os módulos Módulo de segurança Linux (LSM) Lockdown e `yama` também estão habilitados.

`/proc/kcore`

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

Randomização do deslocamento da pilha do kernel na entrada do syscall

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. Isso pode ser ativado configurando `randomize_kstack_offset=on` na linha de comando do kernel.

Verificações de contagem de referência (**`CONFIG_REFCOUNT_FULL`**)

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. No momento, essa opção não está habilitada devido ao possível impacto no desempenho.

Conhecimento do programador sobre SMT núcleos (**CONFIG_SCHED_CORE**)

O kernel AL2 023 é construído com`CONFIG_SCHED_CORE`, o que permite o uso de aplicativos de espaço de usuário. `prctl(PR_SCHED_CORE)` Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Verifique se há corrupção na pilha em chamadas para **schedule()** (**CONFIG_SCHED_STACK_END_CHECK**)

O kernel AL2 023 é construído com `CONFIG_SCHED_STACK_END_CHECK` enabled. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Fortalecimento do alocador de memória

O kernel AL2 023 permite o fortalecimento do alocador de memória do kernel com as opções, e. `CONFIG_SHUFFLE_PAGE_ALLOCATOR` `CONFIG_SLAB_FREELIST_HARDENED` `CONFIG_SLAB_FREELIST_RANDOM` Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

SLUB suporte de depuração

O kernel AL2 023 é ativado, `CONFIG_SLUB_DEBUG` pois essa opção ativa recursos opcionais de depuração para o alocador que podem ser ativados na linha de comando do kernel. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

CONFIG_STATIC_USERMODEHELPER

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP. Isso ocorre porque `CONFIG_STATIC_USERMODEHELPER` requer suporte especial da distribuição que atualmente não está presente no Amazon Linux.

Texto do kernel somente para leitura e rodata (**CONFIG_STRICT_KERNEL_RWX**e**CONFIG_STRICT_MODULE_RWX**)

O kernel AL2 023 é configurado para marcar o texto do kernel e do módulo do kernel e rodata memória somente para leitura e memória não textual marcada como não executável. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

TCP suporte syncookie () **CONFIG_SYN_COOKIES**

O kernel AL2 023 é construído com suporte para cookies de sincronização TCP. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Pilha virtualmente mapeada com páginas de proteção (**CONFIG_VMAP_STACK**)

O kernel AL2 023 é construído com`CONFIG_VMAP_STACK`, permitindo pilhas de kernel mapeadas virtualmente com páginas de proteção. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Crie com avisos do compilador como erros (**CONFIG_WERROR**)

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

Registre a zeragem na função exit (**CONFIG_ZERO_CALL_USED_REGS**)

Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

Endereço mínimo para alocação de espaço de usuário

Essa opção de fortalecimento pode ajudar a reduzir o impacto dos bugs do ponteiro NULL do kernel. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

clang opções específicas de fortalecimento

O kernel AL2 023 é construído com GCC em vez de clang, portanto, a opção de `CONFIG_CFI_CLANG` endurecimento não pode ser ativada, o que também `CONFIG_CFI_PERMISSIVE` não é aplicável. Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

Opções de fortalecimento de kernel específicas do x86-64

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_AMD_IOMMU	N/D	y	N/D	y
CONFIG_AMD_IOMMU_V2	N/D	y	N/D	N/D
CONFIG_IA32_EMULATION	N/D	y	N/D	y
CONFIG_INTEL_IOMMU	N/D	y	N/D	y
CONFIG_INTEL_IOMMU_DEFAULT_ON	N/D	n	N/D	n
CONFIG_INTEL_IOMMU_SVM	N/D	n	N/D	n
CONFIG_LEGACY_VSYSCALL_NONE	N/D	n	N/D	n
CONFIG_MODIFY_LDT_SYSCALL	N/D	n	N/D	n
CONFIG_PAGE_TABLE_ISOLATION	N/D	y	N/D	N/D

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_RANDOMIZE_MEMORY	N/D	y	N/D	y
CONFIG_X86_64	N/D	y	N/D	y
CONFIG_X86_MSR	N/D	y	N/D	y
CONFIG_X86_VSYSCALL_EMULATION	N/D	y	N/D	y
CONFIG_X86_X32	N/D	N/D	N/D	N/D
CONFIG_X86_X32_ABI	N/D	n	N/D	n

Suporte para x86-64

O suporte básico para x86-64 inclui o suporte a bits Physical Address Extension (PAE) e no-execute (NX). Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Suporte para AMD e Intel IOMMU

O kernel AL2 023 é construído com suporte para AMD e Intel IOMMUs. Essa opção é uma das configurações recomendadas do Kernel Self Protection Project.

A opção `CONFIG_INTEL_IOMMU_DEFAULT_ON` não está definida, mas pode ser ativada passando `intel_iommu=on` para a linha de comando do kernel. Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

Atualmente, a `CONFIG_INTEL_IOMMU_SVM` opção não está habilitada em AL2 023. Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

Support para espaço de usuário de 32 bits

> **Important**
>
> O suporte para espaço de usuário x86 de 32 bits está obsoleto e o suporte para execução de binários de espaço de usuário de 32 bits pode ser removido em uma futura versão principal do Amazon Linux.

> **Note**
>
> Embora o AL2 023 não inclua mais pacotes de 32 bits, o kernel ainda suportará a execução de espaço de usuário de 32 bits. Consulte Pacotes x86 (i686) de 32 bits para obter mais informações.

Para oferecer suporte à execução de aplicativos de espaço de usuário de 32 bits, o AL2 023 não ativa a `CONFIG_X86_VSYSCALL_EMULATION` opção e ativa as opções `CONFIG_IA32_EMULATIONCONFIG_COMPAT`, e. `CONFIG_X86_VSYSCALL_EMULATION` Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

A ferramenta x32 A ABI nativa de 32 bits para processadores de 64 bits não está habilitada (`CONFIG_X86_X32eCONFIG_X86_X32_ABI`). Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Registro específico do modelo x86 (MSR) suporte

A opção `CONFIG_X86_MSR` está ativada para oferecer suporte `turbostat`. Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

`modify_ldt` syscall

AL2O 023 não permite que programas de usuário modifiquem a tabela de descritores locais (LDT) x86 com a syscall. `modify_ldt` Essa chamada é necessária para executar código segmentado ou de 16 bits, e sua ausência pode interromper o softwaredosemu, como a execução de alguns programas em WINEe algumas bibliotecas de threading muito antigas. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Remover o mapeamento do kernel no modo de usuário

AL2023 configura o kernel para que a maioria dos endereços do kernel não seja mapeada no espaço do usuário. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Randomize seções de memória do kernel

AL2023 configura o kernel para randomizar os endereços virtuais básicos das seções de memória do kernel. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Opções de endurecimento de kernel específicas do aarch64

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_ARM64_BTI	y	N/D	y	N/D
CONFIG_ARM64_BTI_KERNEL	N/D	N/D	N/D	N/D
CONFIG_ARM64_PTR_AUTH	y	N/D	y	N/D
CONFIG_ARM64_PTR_AUTH_KERNEL	y	N/D	y	N/D

Opção do **CONFIG**	AL2023/6.1/ aarch64	AL2023/6,1/ x86_64	AL2023/6.12/ aarch64	AL2023/6.12/ x86_64
CONFIG_ARM64_SW_TTBR0_PAN	y	N/D	y	N/D
CONFIG_UNMAP_KERNEL_AT_EL0	y	N/D	y	N/D

Identificação do alvo da filial

O kernel AL2 023 permite suporte para Branch Target Identification ()CONFIG_ARM64_BTI. Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

A CONFIG_ARM64_BTI_KERNEL opção não está habilitada no AL2 023, pois é construída com GCC, e o suporte para construir o kernel com essa opção está atualmente desabilitado no kernel upstream devido a um bug do gcc. Embora essa opção seja uma das configurações recomendadas do Kernel Self Protection Project (KSPP), o AL2 023 não define essa opção de configuração conforme recomendado pelo KSPP.

Autenticação de ponteiro (**CONFIG_ARM64_PTR_AUTH**)

O kernel AL2 023 é construído com suporte para a extensão Pointer Authentication (parte das extensões ARMv8 .3), que pode ser usada para ajudar a mitigar as técnicas de Programação Orientada ao Retorno (ROP). O suporte de hardware necessário para autenticação de ponteiro no Graviton foi introduzido com o Graviton 3.

A opção CONFIG_ARM64_PTR_AUTH está habilitada e comporta autenticação de ponteiro no espaço do usuário. Como a CONFIG_ARM64_PTR_AUTH_KERNEL opção também está ativada, o kernel AL2 023 é capaz de usar a proteção do endereço de retorno para si mesmo.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Emule o acesso privilegiado, nunca usando a comutação `TTBR0_EL1`

Essa opção impede que o kernel acesse diretamente a memória do espaço do usuário, com `TTBR0_EL1` sendo definido apenas temporariamente como um valor válido pelas rotinas de acesso do usuário.

Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Desmapear o kernel ao executar no espaço do usuário

O kernel AL2 023 está configurado para desmapear o kernel ao ser executado em userspace (). `CONFIG_UNMAP_KERNEL_AT_EL0` Essa opção é uma das Configurações recomendadas do Kernel Self Protection Project.

Inicialização segura UEFI em 023 AL2

AL2023 suporta UEFI Secure Boot a partir da versão 2023.1. Você deve usar AL2 023 com EC2 instâncias da Amazon que suportam UEFI e UEFI Secure Boot. Para obter mais informações, consulte Requisitos para iniciar uma EC2 instância da Amazon no modo de inicialização UEFI no Guia do EC2 usuário da Amazon.

AL2As instâncias 023 com o UEFI Secure Boot habilitado aceitam somente o código no nível do kernel, incluindo o kernel Linux e os módulos, assinados por Amazon para que você possa garantir que sua instância execute apenas códigos no nível do kernel assinados por AWS.

Para obter mais informações sobre EC2 instâncias da Amazon e UEFI Secure Boot, consulte UEFI Secure Boot para EC2 instâncias da Amazon Amazon no Guia EC2 do usuário da Amazon.

Pré-requisitos

- Você deve usar uma AMI com a versão AL2 023 2023.1 ou superior.
- O tipo de instância deve permitir a inicialização segura do UEFI. Para obter mais informações, consulte Requisitos para iniciar uma EC2 instância da Amazon no modo de inicialização UEFI no Guia do EC2 usuário da Amazon.

Ative a inicialização segura UEFI em 023 AL2

O padrão AL2 023 AMIs incorpora um bootloader e um kernel assinado por nossas chaves. Você pode ativar o UEFI Secure Boot inscrevendo instâncias existentes ou criando AMIs com o UEFI

Secure Boot pré-ativado registrando uma imagem de um snapshot. O UEFI Secure Boot não está habilitado por padrão no padrão AL2 AMIs 023.

O modo de inicialização de AL2 023 AMIs é definido para garantir `uefi-preferred` que as instâncias executadas com elas AMIs usem o firmware UEFI, se o tipo de instância for compatível com UEFI. Se o tipo de instância não for compatível com UEFI, a instância será iniciada com firmware de BIOS antigo. Quando uma instância é executada no modo BIOS antigo, o UEFI Secure Boot não é aplicado.

Para obter mais informações sobre os modos de inicialização da AMI nas EC2 instâncias da Amazon, consulte Comportamento de inicialização da instância com os modos de EC2 inicialização da Amazon Amazon no Guia EC2 do usuário da Amazon.

Tópicos

Inscrição de uma instância existente

Para registrar uma instância existente, preencha as variáveis específicas do firmware UEFI com um conjunto de chaves que permitem que o firmware verifique o carregador de inicialização e o carregador de inicialização verifique o kernel na próxima inicialização.

1. O Amazon Linux fornece uma ferramenta para simplificar o processo de inscrição. Execute o comando a seguir para provisionar a instância com o conjunto necessário de chaves e certificados.

   ```
   sudo amazon-linux-sb enroll
   ```

2. Execute o seguinte comando para reiniciar a instância do . Depois que a instância for reinicializada, o UEFI Secure Boot será ativado.

   ```
   sudo reboot
   ```

> **Note**
>
> AMIs No momento, o Amazon Linux não oferece suporte ao Nitro Trusted Platform Module (NitroTPM). Se você precisar do NitroTPM além do UEFI Secure Boot, use as informações na seção a seguir.

Registrar imagem do instantâneo

Ao registrar uma AMI a partir de um snapshot de um volume raiz do Amazon EBS usando a EC2 `register-image` API da Amazon, você pode provisionar a AMI com um blob binário que contém o estado do armazenamento de variáveis UEFI. Ao fornecer o AL2 023`UefiData`, você ativa o UEFI Secure Boot e não precisa seguir as etapas na seção anterior.

Para obter mais informações sobre como criar e usar um blob binário, consulte Criar um blob binário contendo um armazenamento de variáveis pré-preenchido no Guia do usuário da Amazon EC2 .

AL2O 023 fornece um blob binário pré-criado que pode ser usado diretamente nas instâncias da Amazon. EC2 O blob binário está localizado em `/usr/share/amazon-linux-sb-keys/uefi.vars` em uma instância em execução. Esse blob é fornecido pelo pacote `amazon-linux-sb-keys` RPM, que é instalado por padrão em AL2 023 a AMIs partir da versão 2023.1.

> **Note**
>
> Para garantir que você esteja usando a versão mais recente das chaves e revogações, use o blob da mesma versão de AL2 023 que você usa para criar a AMI.

Ao registrar uma imagem, recomendamos usar o parâmetro `BootMode` da API `RegisterImage` definida como `uefi`. Isso permite que você ative o NitroTPM definindo o parâmetro `TpmSupport` como `v2.0`. Além disso, definir `BootMode` para `uefi` garantir que o UEFI Secure Boot esteja habilitado e não possa ser desativado acidentalmente ao mudar para um tipo de instância que não seja compatível com UEFI.

Para obter mais informações sobre o NitroTPM, consulte NitroTPM para instâncias da Amazon EC2 Amazon no Guia do usuário da Amazon EC2 .

Atualizações de revogação

Talvez seja necessário que o Amazon Linux distribua uma nova versão do bootloader `grub2` ou do kernel Linux assinado com chaves atualizadas. Nesse caso, talvez seja necessário revogar a chave antiga para evitar a chance de permitir que bugs exploráveis de versões anteriores do bootloader ignorem o processo de verificação do UEFI Secure Boot.

As atualizações de pacotes do `grub2` ou `kernel` sempre atualizam automaticamente a lista de revogações no armazenamento de variáveis UEFI da instância em execução. Isso significa que, com o UEFI Secure Boot ativado, você não pode mais executar a versão antiga de um pacote depois de instalar uma atualização de segurança para o pacote.

Como o UEFI Secure Boot funciona em 023 AL2

Ao contrário de outras distribuições Linux, o Amazon Linux não fornece um componente adicional, chamado shim, para atuar como o bootloader de primeiro estágio. O calço geralmente é assinado com chaves da Microsoft. Por exemplo, em distribuições Linux com o shim, o shim carrega o bootloader `grub2` que usa o próprio código do shim para verificar o kernel Linux. Além disso, o shim mantém seu próprio conjunto de chaves e revogações no banco de dados da Machine Owner Key (MOK) localizado no armazenamento de variáveis UEFI e controlado com a ferramenta `mokutil`.

O Amazon Linux não oferece nada. Como o proprietário da AMI controla as variáveis da UEFI, essa etapa intermediária não é necessária e afetaria negativamente os tempos de lançamento e inicialização. Além disso, optamos por não incluir a confiança em nenhuma chave de fornecedor por padrão, para reduzir a chance de binários indesejados serem executados. Como sempre, os clientes podem incluir binários se quiserem fazer isso.

Com o Amazon Linux, o UEFI carrega e verifica diretamente nosso `grub2` bootloader. O bootloader `grub2` foi modificado para usar UEFI para verificar o kernel Linux após carregá-lo. Assim, o Kernel Linux é verificado usando os mesmos certificados armazenados na variável UEFI normal `db` (banco de dados de chaves autorizado) e testado com a mesma variável `dbx` (banco de dados de revogações) do bootloader e outros binários UEFI. Como fornecemos nossas próprias chaves PK e KEK, que controlam o acesso ao banco de dados db e ao banco de dados dbx, podemos distribuir atualizações e revogações assinadas conforme necessário, sem um intermediário como o shim.

Para obter mais informações sobre o UEFI Secure Boot, consulte Como o UEFI Secure Boot funciona com EC2 instâncias da Amazon Amazon no Guia EC2 do usuário da Amazon.

Inscrevendo suas próprias chaves

Conforme documentado na seção anterior, o Amazon Linux não exige uma inicialização segura `shim` para UEFI na Amazon EC2. Ao ler a documentação de outras distribuições Linux, você pode encontrar documentação para gerenciar o banco de dados Machine Owner Key (MOK) usando`mokutil`, que não está presente em AL2 023. Os ambientes `shim` e MOK contornam algumas limitações de registro de chaves no firmware UEFI que não são aplicáveis à forma como a Amazon EC2 implementa o UEFI Secure Boot. Com a Amazon, EC2 existem mecanismos para manipular facilmente as chaves no armazenamento de variáveis UEFI.

Se quiser registrar suas próprias chaves, você pode fazer isso manipulando o armazenamento de variáveis em uma instância existente (consulte Adicionar chaves ao armazenamento de variáveis de dentro da instância) ou criando um blob binário pré-preenchido (consulte Criar um blob binário contendo um armazenamento de variáveis pré-preenchido).

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.