aws

Guide de l’utilisateur

Amazon Linux 2023

Amazon Linux 2023: Guide de l’utilisateur

Table of Contents

Qu'est-ce qu'Amazon Linux 2023 ?

Amazon Linux 2023 (AL2023) est la nouvelle génération d'Amazon Linux d'Amazon Web Services (AWS). Avec AL2023, vous pouvez développer et exécuter des applications cloud et d'entreprise dans un environnement d'exécution sécurisé, stable et performant. Vous bénéficiez également d'un environnement applicatif offrant un support à long terme avec un accès aux dernières innovations de Linux. AL2023 est fourni sans frais supplémentaires.

AL2023 est le successeur d'Amazon Linux 2 (AL2). Pour plus d'informations sur les différences entre AL2023 et AL2, voir et Modifications apportées aux Comparaison entre AL2 et AL2023 packages dans AL2023.

Rubriques

Cadence de publication

Une nouvelle version majeure d'Amazon Linux est publiée tous les deux ans et inclut cinq ans de support. Chaque version inclut un support en deux phases. La phase de support standard couvre les deux premières années. Ensuite, une phase de maintenance permet de poursuivre le support pendant trois années supplémentaires.

Dans la phase de support standard, la version reçoit des mises à jour mineures trimestrielles. Pendant la phase de maintenance, une version reçoit uniquement les mises à jour de sécurité et les corrections de bogues critiques publiées dès leur publication.

Année	Amazon Linux 2023	Amazon Linux 2025	Amazon Linux 2027	Amazon Linux 2029
2023	 Support standard			
2024	 Support standard			
2025	Maintenance	 Support standard		
2026	Maintenance	 Support standard		
2027	Maintenance	Maintenance	 Support standard	
2028	 Fin de vie	Maintenance	 Support standard	
2029	 Fin de vie	Maintenance	Maintenance	 Support standard

Année	Amazon Linux 2023	Amazon Linux 2025	Amazon Linux 2027	Amazon Linux 2029
2030	 Fin de vie	 Fin de vie	Maintenance	 Support standard
2031	 Fin de vie	 Fin de vie	Maintenance	Maintenance

Versions majeures et mineures

À chaque version d'Amazon Linux (version majeure, version mineure ou version de sécurité), nous publions une nouvelle Amazon Machine Image (AMI) pour Linux.

- Version majeure : inclut de nouvelles fonctionnalités et des améliorations en matière de sécurité et de performances dans l'ensemble de la pile. Les améliorations peuvent inclure des modifications majeures du noyau, de la chaîne d'outils, de Glib C, d'OpenSSL, et de toutes les autres bibliothèques et utilitaires du système. Les versions majeures d'Amazon Linux sont basées en partie sur la version actuelle de la distribution Fedora Linux en amont. AWS peut ajouter ou remplacer des packages spécifiques provenant d'autres distributions en amont que Fedora.
- Version mineure : mise à jour trimestrielle qui inclut des mises à jour de sécurité, des correctifs de bogues, ainsi que des nouvelles fonctionnalités et de nouveaux packages. Chaque version mineure est une liste cumulative de mises à jour qui inclut des correctifs de sécurité et de bogues, ainsi que de nouvelles fonctionnalités et de nouveaux packages. Ces versions peuvent inclure les dernières exécutions des langues, comme PHP. Elles peuvent également inclure d'autres packages logiciels populaires comme Ansible et Docker.

Utilisation de nouvelles versions

Les mises à jour sont fournies par le biais d'une combinaison des nouvelles versions d'Amazon Machine Image (AMI) et des nouveaux référentiels correspondants. Par défaut, une nouvelle AMI et le référentiel vers lequel elle pointe sont couplés. Cependant, vous pouvez au fil du temps rediriger vos instances Amazon EC2 en cours d'exécution vers de nouvelles versions de référentiel afin

d'appliquer les mises à jour aux instances en cours d'exécution. Vous pouvez également effectuer une mise à jour en lançant de nouvelles instances des dernières AMI.

Politique de support à long terme

Amazon Linux fournit des mises à jour pour tous vos packages et assure la compatibilité au sein d'une version majeure pour vos applications basées sur Amazon Linux. Les packages de base tels que la bibliothèque glibc, OpenSSL, OpenSSH et le gestionnaire de packages DNF reçoivent un support pendant toute la durée de vie de la version majeure d'AL2023. Les packages qui ne font pas partie des packages de base sont pris en charge en fonction de leurs sources en amont spécifiques. Vous pouvez consulter le statut et les dates de prise en charge spécifiques de chaque package en exécutant la commande suivante.

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

Vous pouvez obtenir des informations sur tous les packages actuellement installés en exécutant la commande suivante.

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

La liste complète des packages de base est finalisée lors de la prévisualisation. Si vous souhaitez voir d'autres packages inclus en tant que packages de base, dites-le-nous. Nous évaluons à mesure que nous collectons des commentaires. Les commentaires sur AL2023 peuvent être fournis par l'intermédiaire de votre représentant AWS désigné ou en signalant un problème dans le référentiel amazon-linux-2023 sur GitHub.

Dénomination et gestion des versions

AL2023 fournit une version mineure tous les trois mois pendant les deux années de support standard. Chaque version est identifiée par un incrément compris entre 0 et N. 0 fait référence à la version principale d'origine pour cette itération. Toutes les versions s'appellent Amazon Linux 2023. Lors de la sortie d'Amazon Linux 2025, AL2023 bénéficie d'un support étendu et reçoit des mises à jour pour les mises à jour de sécurité et les correctifs de bogues critiques.

Par exemple, les versions mineures d'AL2023 ont le format suivant :

- `2023.0.20230301`

- `2023.1.20230601`
- `2023.2.20230901`

Les AMI AL2023 correspondantes ont le format suivant :

- `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`

Dans une version mineure spécifique, les versions régulières de l'AMI ont lieu avec un horodatage de la date de publication de l'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`

La méthode recommandée pour identifier une instance AL2 ou AL2023 commence par la lecture de la chaîne CPE (Common Platform Enumeration) à partir de. `/etc/system-release-cpe` Divisez ensuite la chaîne en ses champs. Enfin, lisez les valeurs de la plateforme et de la version.

AL2023 introduit également de nouveaux fichiers pour l'identification des plateformes :

- Liens symboliques `/etc/amazon-linux-release` vers `/etc/system-release`
- Liens symboliques `/etc/amazon-linux-release-cpe` vers `/etc/system-release-cpe`

Ces deux fichiers indiquent qu'une instance est Amazon Linux. Il n'est pas nécessaire de lire un fichier ou de diviser la chaîne en champs, sauf si vous souhaitez connaître les valeurs spécifiques de la plateforme et de la version.

Optimisations des performances et des opérations

Noyau Amazon Linux 6.1

- L'AL2023 utilise les derniers pilotes pour les appareils Elastic Network Adapter (ENA) et Elastic Fabric Adapter (EFA). L'AL2023 se concentre sur les rétroportages de performances et de fonctionnalités pour le matériel de l'infrastructure Amazon EC2.

- Les correctifs à chaud du noyau sont disponibles pour les types d'instance `x86_64` et `aarch64`. Cela réduit la nécessité de redémarrages fréquents.
- Toutes les configurations de construction et d'exécution du noyau incluent de nombreuses optimisations opérationnelles et de performance identiques à celles d'AL2.

Sélection de la chaîne d'outils de base et indicateurs de création par défaut

- Les packages AL2023 sont créés avec les optimisations du compilateur (`-O2`) activées par défaut
- Les packages AL2023 sont créés en nécessitant `x86-64v2` pour les systèmes `x86-64` (`-march=x86-64-v2`), et Graviton 2 ou supérieur pour `aarch64` (`-march=armv8.2-a+crypto -mtune=neoverse-n1`).
- Les packages AL2023 sont créés avec la vectorisation automatique activée (`-ftree-vectorize`).
- Les packages AL2023 sont créés avec l'optimisation du temps de liaison (LTO) activée.
- AL2023 utilise les versions mises à jour de Rust, Clang/LLVM et Go.

Sélection du package et versions

- Certains rétroportages vers les principaux composants du système incluent plusieurs améliorations de performances pour l'exécution sur l'infrastructure Amazon EC2, en particulier les instances Graviton.
- L'AL2023 est intégré à plusieurs Services AWS fonctionnalités. Cela inclut l' AWS CLI agent SSM, l'agent Amazon Kinesis et. CloudFormation
- AL2023 utilise Amazon Corretto comme kit de développement Java (JDK).
- AL2023 fournit des moteurs de base de données et des mises à jour d'exécution de langage de programmation aux nouvelles versions à mesure qu'elles sont publiées par des projets en amont. Les exécutions de langage de programmation dotées de nouvelles versions sont ajoutées lors de leur publication.

Déploiement dans un environnement cloud

- L'AMI AL2023 de base et les images de conteneurs sont fréquemment mises à jour pour prendre en charge le remplacement des instances par correctifs.

- Les mises à jour du noyau sont incluses dans les mises à jour de l'AMI AL2023. Cela signifie que vous n'avez pas besoin d'utiliser des commandes telles que `yum update` et `reboot` pour mettre à jour le noyau.
- Outre l'AMI AL2023 standard, une AMI minimale et une image de conteneur sont également disponibles. Choisissez l'AMI minimale pour exécuter un environnement avec le nombre minimal de packages requis pour exécuter le service.
- Par défaut, les AMI et les conteneurs AL2023 sont verrouillés sur une version spécifique des référentiels de packages. Il n'y a pas de mise à jour automatique lors de leur lancement. Cela signifie que vous avez toujours le contrôle du moment où vous ingérez une mise à jour de package. Vous pouvez toujours effectuer des tests dans un environnement bêta/gamma avant de passer à la production. En cas de problème, vous pouvez utiliser le chemin de restauration prévalidé.

Relation avec Fedora

AL2023 maintient ses propres cycles de vie de publication et de support indépendamment de Fedora. AL2023 fournit des versions mises à jour de logiciels open source, une plus grande variété de packages et des sorties fréquentes. Cela préserve les systèmes d'exploitation habituels basés sur le RPM.

La version généralement disponible (GA) d'AL2023 n'est pas directement comparable à une version spécifique de Fedora. La version GA d'AL2023 inclut des composants de Fedora 34, 35 et 36. Certains composants sont identiques à ceux de Fedora et d'autres sont modifiés. D'autres composants ressemblent davantage à ceux de CentOS 9 Streams ou ont été développés indépendamment. Le noyau Amazon Linux provient des options de support à long terme disponibles sur kernel.org, choisies indépendamment de Fedora.

Version cloud-init personnalisée

Le package cloud-init est une application open source qui amorce les images Linux dans un environnement de cloud computing. Pour plus d'informations, consultez la documentation cloud-init.

AL2023 contient une version personnalisée de cloud-init. Avec cloud-init, vous pouvez spécifier ce qui arrive à votre instance au moment du démarrage.

Lorsque vous lancez une instance, vous pouvez utiliser les champs de données utilisateur pour transmettre des actions àcloud-init. Cela signifie que vous pouvez utiliser des AMI (Amazon Machine

Image) communes pour de nombreux cas d'utilisation et les configurer dynamiquement quand vous démarrez une instance. AL2023 utilise également cloud-init pour configurer le compte `ec2-user`.

AL2023 utilise les actions cloud-init dans `/etc/cloud/cloud.cfg.d` et `/etc/cloud/cloud.cfg`. Vous pouvez créer vos propres fichiers d'actions cloud-init dans le répertoire `/etc/cloud/cloud.cfg.d`. Cloud-init lit tous les fichiers de ce répertoire dans l'ordre lexicographique. Les fichiers ultérieurs remplacent les valeurs des fichiers plus anciens. Quand cloud-init lance une instance, le package cloud-init effectue les tâches de configuration suivantes :

- Il définit les paramètres régionaux par défaut.
- Il définit le nom d'hôte.
- Il analyse et gère les données utilisateur.
- Il génère des clés SSH privées d'hôte.
- Il ajoute les clés SSH publiques d'un utilisateur à `.ssh/authorized_keys` pour faciliter la connexion et l'administration.
- Il prépare les référentiels pour la gestion des packages.
- Il traite les actions de package définies dans les données utilisateur.
- Il exécute les scripts utilisateur qui se trouvent dans les données utilisateur.
- Il monte les volumes de stockage d'instances, le cas échéant.
  - Par défaut, si le volume de stockage d'instances `ephemeral0` est présent et contient un système de fichiers valide, le volume de stockage d'instances est monté dans `/media/ephemeral0`. Sinon, il n'est pas monté.
  - Par défaut, pour les types d'instance `m1.small` et `c1.medium`, tous les volumes d'échange associés à l'instance sont montés.
  - Vous pouvez remplacer le montage de volume de stockage d'instances par défaut avec la directive cloud-init suivante :

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

    Pour plus de contrôle sur les montages, consultez Montages (langue française non garantie) dans la documentation cloud-init.

- Lorsqu'une instance est lancée, les volumes de stockage d'instances qui prennent en charge TRIM ne sont pas formatés. Avant de monter des volumes de stockage d'instances, vous devez partitionner et formater les volumes de stockage d'instances.

  Pour plus d'informations, consultez la section Support TRIM du volume de stockage d'instance dans le guide de l'utilisateur Amazon EC2.
- Lorsque vous lancez vos instances, vous pouvez utiliser le module `disk_setup` pour partitionner et formater vos volumes de stockage d'instances.

  Pour plus d'informations, consultez Configuration de disque (langue française non garantie) dans la documentation cloud-init.

Pour en savoir plus sur l'utilisation de cloud-init avec SELinux, consultez Utilisation de cloud-init pour activer le mode `enforcing`.

Pour en savoir plus sur les formats de données utilisateur cloud-init, consultez Formats de données utilisateur dans la documentation cloud-init.

Mises à jour et fonctionnalités de sécurité

AL2023 fournit de nombreuses mises à jour et solutions de sécurité.

Rubriques

- Gestion des mises à jour
- Sécurité dans le cloud
- Modes SELinux
- Programme de conformité
- Serveur SSH par défaut
- Principales fonctionnalités d'OpenSSL 3

Gestion des mises à jour

Appliquez les mises à jour de sécurité à DNF l'aide des versions du référentiel. Pour plus d'informations, consultez Gérez les mises à jour des packages et du système d'exploitation dans AL2023.

Sécurité dans le cloud

La sécurité est une responsabilité partagée entre vous AWS et vous. Le modèle de responsabilité partagée décrit cela comme la sécurité du cloud et la sécurité dans le cloud. Pour plus d'informations, consultez Sécurité et conformité dans Amazon Linux 2.

Modes SELinux

Par défaut, SELinux est activé en mode permissif dans AL2023. En mode permissif, les refus d'autorisation sont journalisés mais ne sont pas appliqués.

Les politiques SELinux définissent les autorisations pour les utilisateurs, les processus, les programmes, les fichiers et les appareils. Avec SELinux, vous pouvez choisir l'une des deux politiques. Les politiques sont ciblées ou de sécurité multi-niveaux (MLS).

Pour plus d'informations sur les modes et la politique SELinux, consultez Configuration des modes SELinux pour AL2023 et le wiki du projet SELinux.

Programme de conformité

Des auditeurs indépendants évaluent la sécurité et la conformité de la norme AL2023 ainsi que de nombreux programmes de AWS conformité.

Serveur SSH par défaut

AL2023 inclut OpenSSH 8.7. OpenSSH 8.7 désactive par défaut l'algorithme d'échange de clés `ssh-rsa`. Pour plus d'informations, consultez Configuration du serveur SSH par défaut.

Principales fonctionnalités d'OpenSSL 3

- Le protocole CMP (Certificate Management Protocol, RFC 4210) inclut à la fois le CRMF (RFC 4211) et le transfert HTTP (RFC 6712).
- Un client HTTPS ou HTTP dans libcrypto prend en charge les actions GET et POST, la redirection, le contenu brut et encodé en ASN.1, les proxys et les délais d'expiration.
- EVP_KDF fonctionne avec les fonctions de dérivation de clé.
- EVP_MAC API fonctionne avec MACs.
- Prise en charge du noyau Linux TLS.

Pour plus d'informations, consultez le Guide de migration d'OpenSSL.

Service de mise en réseau

Le projet open source `systemd-networkd` est largement disponible dans les distributions Linux modernes. Le projet utilise un langage de configuration déclaratif similaire au reste du framework `systemd`. Ses principaux types de fichiers de configuration sont les fichiers `.link` et `.network`.

Le package `amazon-ec2-net-utils` génère des configurations spécifiques à l'interface dans le répertoire `/run/systemd/network`. Ces configurations permettent de mettre en réseau IPv4 et IPv6 sur les interfaces lorsqu'elles sont associées à une instance. Ces configurations installent également des règles de routage de stratégie qui permettent de garantir que le trafic d'origine locale est acheminé vers le réseau via l'interface réseau de l'instance correspondante. Ces règles garantissent que le trafic approprié est acheminé via l'Elastic Network Interface (ENI) à partir des adresses ou des préfixes associés. Pour plus d'informations sur l'utilisation d'ENI, consultez la section Utilisation d'ENI dans le guide de l'utilisateur Amazon EC2.

Vous pouvez personnaliser ce comportement réseau en plaçant un fichier de configuration personnalisé dans le répertoire `/etc/systemd/network` afin de remplacer les paramètres de configuration par défaut contenus dans `/run/systemd/network`.

La documentation systemd.network décrit comment le service `systemd-networkd` détermine la configuration qui s'applique à une interface spécifique. Il génère également des noms alternatifs, appelésaltnames, pour les interfaces basées sur l'ENI afin de refléter les propriétés des différentes AWS ressources. Ces propriétés d'interface soutenues par l'ENI sont l'`ENI ID` et le champ `DeviceIndex` de la pièce jointe ENI. Vous pouvez faire référence à ces interfaces à l'aide de leurs propriétés lorsque vous utilisez différents outils comme la commande `ip`.

Les noms d'interface d'instance AL2023 sont générés à l'aide du schéma de dénomination des `systemd` emplacements. Pour plus d'informations, consultez schéma de dénomination systemd.net.

De plus, AL2023 utilise par défaut l'algorithme actif de planification des transmissions du réseau de gestion des files d'attente, `fq_codel`. Pour plus d'informations, reportez-vous à la section CoDelVue d'ensemble.

Packages de chaîne d'outils de base glibc, gcc, binutils

Dans Amazon Linux, un sous-ensemble de packages est désigné sous le nom de packages de chaîne d'outils de base. Dans le cadre de l'AL2023, les packages de base bénéficient d'un support

de cinq ans. Nous pouvons modifier la version d'un package, mais la prise en charge à long terme s'applique au package inclus dans la version d'Amazon Linux.

Ces trois packages de base fournissent une chaîne d'outils système utilisée pour créer la plupart des logiciels dans la distribution Amazon Linux.

Package	Définition	Objectif
glibc 2.34	Bibliothèque système C	Utilisée par la plupart des programmes binaires fournissant des fonctions standard et par l'interface entre les programmes et le noyau.
gcc 11.2	Suite de compilateurs gcc	Compile C, C++, Fortran.
binutils 2.35	Outils d'assemblage et de liaison, ainsi que d'autres outils binaires	Manipule ou inspecte des programmes binaires.

Nous recommandons que les mises à jour de n'importe quelle bibliothèque glibc soient suivies d'un redémarrage. Pour les mises à jour des packages qui contrôlent les services, il peut s'avérer suffisant de redémarrer les services pour récupérer les mises à jour. Cependant, un redémarrage du système garantit que toutes les mises à jour précédentes du package et de la bibliothèque sont terminées.

Outil de gestion de package

L'outil de gestion des packages logiciels par défaut dans AL2023 estDNF. DNFest le successeur de YUM l'outil de gestion de packages d'AL2.

DNF est similaire à YUM dans son utilisation. De nombreuses DNF commandes et options de commande sont identiques aux YUM commandes. Dans une commande d'interface de ligne de commande (CLI), `dnf` remplace `yum` dans la plupart des cas.

Par exemple, pour les `yum` commandes AL2 suivantes :

```
$ sudo yum install packagename
```

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

Dans AL2023, elles deviennent les commandes suivantes :

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

Dans AL2023, la commande `yum` est toujours disponible, mais en tant que pointeur vers la commande `dnf`. Ainsi, lorsque la commande `yum` est utilisée dans le shell ou dans un script, toutes les commandes et options sont identiques à DNF CLI. Pour plus d'informations sur les différences entre la YUM CLI et la DNF CLI, consultez Modifications dans DNF CLI par rapport à YUM.

Pour une référence complète des commandes et des options associées à la commande `dnf`, reportez-vous à la page du man `man dnf`. Pour plus d'informations, consultez la section Référence des DNF commandes.

Configuration du serveur SSH par défaut

Si vous avez des clients SSH datant d'il y a plusieurs années, il se peut qu'une erreur s'affiche lorsque vous vous connectez à une instance. Si l'erreur indique qu'aucun type de clé d'hôte correspondant n'a été trouvé, mettez à jour la clé d'hôte SSH pour résoudre ce problème.

Désactivation des signatures **`ssh-rsa`** par défaut

L'AL2023 inclut une configuration par défaut qui désactive l'ancien algorithme de clé `ssh-rsa` d'hôte et génère un ensemble réduit de clés d'hôte. Les clients doivent prendre en charge l'algorithme de clé d'hôte `ssh-ed25519` ou `ecdsa-sha2-nistp256`.

La configuration par défaut accepte n'importe lequel de ces algorithmes d'échange de clés :

- `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`

Par défaut, AL2023 génère les clés d'hôte `ed25519` et `ECDSA`. Les clients prennent en charge l'algorithme de clé d'hôte `ssh-ed25519` ou `ecdsa-sha2-nistp256`. Lorsque vous vous connectez par SSH à une instance, vous devez utiliser un client qui prend en charge un algorithme compatible, comme `ssh-ed25519` ou `ecdsa-sha2-nistp256`. Si vous devez utiliser d'autres types de clés, remplacez la liste des clés générées par un fragment `cloud-config` de données utilisateur.

Dans l'exemple suivant, `cloud-config` génère une clé d'hôte `rsa` avec les clés `ed25519` et `ecdsa`.

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

Si vous utilisez une paire de clés RSA pour l'authentification par clé publique, votre client SSH doit prendre en charge une signature `rsa-sha2-256` ou `rsa-sha2-512`. Si vous utilisez un client incompatible et que vous ne pouvez pas effectuer de mise à niveau, réactivez la prise en charge de `ssh-rsa` sur votre instance. Pour réactiver le `ssh-rsa` support, activez la politique de chiffrement `LEGACY` du système à l'aide des commandes suivantes.

```
$ sudo dnf install crypto-policies-scripts
$ sudo update-crypto-policies --set LEGACY
```

Pour plus d'informations sur la gestion des clés d'hôte, consultez la section Amazon Linux Host keys.

Fonctionnalité obsolète dans AL2023

Les fonctionnalités déconseillées dans AL2 et absentes dans AL2023 sont documentées ici. Il s'agit de fonctionnalités telles que les fonctionnalités et les packages qui sont présentes dans AL2, mais pas dans AL2023 et qui ne seront pas ajoutées à AL2023. Pour plus d'informations sur la durée pendant laquelle la fonctionnalité est prise en charge dans AL2, voir Fonctionnalité obsolète dans AL2.

Il existe également une fonctionnalité dans AL2023 qui est obsolète et sera supprimée dans une future version. Ce chapitre décrit en quoi consiste cette fonctionnalité, à quel moment elle n'est plus prise en charge et à quel moment elle sera supprimée d'Amazon Linux. Comprendre les fonctionnalités obsolètes vous aidera à déployer AL2023 et à préparer la prochaine version majeure d'Amazon Linux.

Rubriques

Packages `compat-`

Tous les packages dans AL2 avec le préfixe de `compat-` sont fournis pour assurer la compatibilité binaire avec les anciens binaires qui n'ont pas encore été reconstruits pour les versions modernes du package. Chaque nouvelle version majeure d'Amazon Linux ne reprendra aucun `compat-` package des versions précédentes.

Tous les `compat-` packages d'une version d'Amazon Linux (par exemple AL2) sont obsolètes et ne sont pas présents dans la version suivante (par exemple AL2023). Nous recommandons vivement de reconstruire le logiciel en fonction des versions mises à jour des bibliothèques.

Fonctionnalité obsolète abandonnée dans AL1, supprimée dans AL2

Cette section décrit les fonctionnalités disponibles dans AL1 et qui ne le sont plus dans AL2.

> **Note**
>
> Dans le cadre de la phase de support de maintenance d'AL1, la date de fin de end-of-life vie de certains packages était antérieure à celle d'AL1. Pour plus d'informations, consultez les déclarations de support du package AL1.

> **Note**
>
> Certaines fonctionnalités AL1 ont été abandonnées dans les versions précédentes. Pour plus d'informations, consultez les notes de mise à jour de l'AL1.

Rubriques

AMI x86 (i686) 32 bits

Dans le cadre de la version 2014.09 d'AL1, Amazon Linux a annoncé qu'il s'agirait de la dernière version à produire des AMI 32 bits. Par conséquent, depuis la version 2015.03 d'AL1, Amazon Linux ne prend plus en charge l'exécution du système en mode 32 bits. AL2 offre un support d'exécution limité pour les binaires 32 bits sur des hôtes x86-64 et ne fournit pas de packages de développement permettant de créer de nouveaux binaires 32 bits. AL2023 n'inclut plus aucun package d'espace utilisateur 32 bits. Nous recommandons aux utilisateurs de terminer leur transition vers le code 64 bits avant de migrer vers AL2023.

Si vous devez exécuter des fichiers binaires 32 bits sur AL2023, il est possible d'utiliser l'espace utilisateur 32 bits d'AL2 dans un conteneur AL2 exécuté au-dessus d'AL2023.

`aws-apitools-*`remplacé par AWS CLI

Avant la sortie du AWS CLI en septembre 2013, AWS a mis à disposition un ensemble d'utilitaires de ligne de commande, implémentés dansJava, qui permettaient aux utilisateurs de passer des appels d'API Amazon EC2. Ces outils ont été abandonnés en 2015, et ils sont AWS CLI devenus le

moyen privilégié d'interagir avec les API Amazon EC2 depuis la ligne de commande. L'ensemble des utilitaires de ligne de commande inclut les `aws-apitools-*` packages suivants.

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

Le support en amont pour les `aws-apitools-*` packages a pris fin en mars 2017. Malgré l'absence de support en amont, Amazon Linux a continué à fournir certains de ces utilitaires de ligne de commande`aws-apitools-ec2`, notamment pour assurer la rétrocompatibilité aux utilisateurs. AWS CLI Il s'agit d'un outil plus robuste et plus complet que les `aws-apitools-*` packages car il est activement maintenu et fournit un moyen d'utiliser toutes les AWS API.

Les `aws-apitools-*` packages sont devenus obsolètes en mars 2017 et ne recevront plus de mises à jour. Tous les utilisateurs de l'un de ces packages doivent migrer vers le AWS CLI dès que possible. Ces packages ne sont pas présents dans AL2023.

AL1 a également fourni les `aws-apitools-rds` packages `aws-apitools-iam` et, qui étaient obsolètes dans AL1, et qui ne sont plus présents dans Amazon Linux à partir d'AL2.

`systemd`remplace **`upstart`** en AL2

AL2 a été la première version d'Amazon Linux à utiliser le système d'`systemd`initialisation, `upstart` en remplacement de AL1. Toute configuration `upstart` spécifique doit être modifiée dans le cadre de la migration d'AL1 vers une version plus récente d'Amazon Linux. Comme il n'est pas possible de l'utiliser `systemd` sur AL1, le passage de `upstart` à ne `systemd` peut être effectué que dans le cadre du passage à une version majeure plus récente d'Amazon Linux, telle que AL2 ou AL2023.

Fonctionnalité déconseillée dans AL2 et supprimée dans AL2023

Cette section décrit les fonctionnalités disponibles dans AL2 et qui ne sont plus disponibles dans AL2023.

Rubriques

Packages x86 (i686) 32 bits

Dans le cadre de la version 2014.09 d'AL1, nous avons annoncé qu'il s'agirait de la dernière version à produire des AMI 32 bits. Par conséquent, depuis la version 2015.03 d'AL1, Amazon Linux ne prend plus en charge l'exécution du système en mode 32 bits. AL2 fournit un support d'exécution limité pour les binaires 32 bits sur des hôtes x86-64 et ne fournit pas de packages de développement permettant de créer de nouveaux binaires 32 bits. AL2023 n'inclut plus aucun package d'espace utilisateur 32 bits. Nous recommandons aux clients de terminer leur transition vers le code 64 bits.

Si vous devez exécuter des fichiers binaires 32 bits sur AL2023, il est possible d'utiliser l'espace utilisateur 32 bits d'AL2 dans un conteneur AL2 exécuté au-dessus d'AL2023.

`aws-apitools-*`remplacé par AWS CLI

Avant la sortie du AWS CLI en septembre 2013, AWS a mis à disposition un ensemble d'utilitaires de ligne de commande, implémentés dansJava, qui permettaient aux clients de passer des appels d'API Amazon EC2. Ces outils ont été déconseillés en 2015 et sont AWS CLI devenus le moyen préféré d'interagir avec les API Amazon EC2 depuis la ligne de commande. Cela inclut les `aws-apitools-*` packages suivants.

- `aws-apitools-as`
- `aws-apitools-cfn`
- `aws-apitools-common`

- `aws-apitools-ec2`
- `aws-apitools-elb`
- `aws-apitools-mon`

Le support en amont pour les `aws-apitools-*` packages a pris fin en mars 2017. Malgré l'absence de support en amont, Amazon Linux a continué à fournir certains de ces utilitaires de ligne de commande (tels que`aws-apitools-ec2`) afin de fournir une rétrocompatibilité aux clients. AWS CLI Il s'agit d'un outil plus robuste et complet que les `aws-apitools-*` packages car il est activement maintenu et fournit un moyen d'utiliser toutes les AWS API.

Les `aws-apitools-*` packages sont devenus obsolètes en mars 2017 et ne recevront plus de mises à jour. Tous les utilisateurs de l'un de ces packages doivent migrer vers le AWS CLI dès que possible. Ces packages ne sont pas présents dans AL2023.

`bzr`système de contrôle de révision

Le système de contrôle de révision GNU Bazaar(`bzr`) est abandonné dans AL2 et n'est plus présent dans AL2023.

Il est `bzr` conseillé aux utilisateurs de migrer leurs référentiels vers`git`.

cgroup v1

AL2023 passe à la hiérarchie des groupes de contrôle unifiés (cgroup v2), tandis qu'AL2 utilise cgroup v1. Comme AL2 ne prend pas en charge cgroup v2, cette migration doit être terminée dans le cadre du passage à AL2023.

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

> **Note**
>
> Le `log4j-cve-2021-44228-hotpatch` package est obsolète dans AL2 et supprimé dans AL2023.

En réponse à CVE-2021-44228, Amazon Linux a publié une version empaquetée RPM du Hotpatch pour Apache Log4j pour AL1 et AL2. Dans l'annonce de l'ajout du hotpatch à Amazon Linux, nous

avons noté que « l'installation du hotpatch ne remplace pas la mise à jour vers une version de log4j qui atténue le CVE-2021-44228 ou le CVE-2021-45046 ».

La mise à jour corrective à chaud était une mesure d'atténuation permettant de laisser le temps nécessaire au correctif `log4j`. La première version de disponibilité générale d'AL2023 a eu lieu 15 mois après CVE-2021-44228. Par conséquent, AL2023 n'est pas livré avec le hotpatch (activé ou non).

Il est conseillé aux clients qui exécutent leurs propres versions `log4j` sur Amazon Linux de s'assurer qu'ils ont effectué une mise à jour vers des versions non affectées par les CVE-2021-44228 et les CVE-2021-45046.

`lsb_release` et le package `system-lsb-core`

Historiquement, certains logiciels invoquaient la commande `lsb_release` (fournie dans AL2 par le package `system-lsb-core`) pour obtenir des informations sur la distribution Linux sur laquelle ils s'exécutaient. Le projet Linux Standards Base (LSB) a introduit cette commande et les distributions Linux l'ont adoptée. Les distributions Linux ont évolué pour utiliser le standard simplifié consistant à conserver ces informations dans `/etc/os-release` et d'autres fichiers connexes.

Le standard `os-release` est issu de `systemd`. Pour plus d'informations, consultez la documentation systemd sur os-release (langue française non garantie).

AL2023 n'est pas fourni avec la commande `lsb_release` et n'inclut pas le package `system-lsb-core`. Le logiciel doit terminer la transition vers le standard `os-release` pour maintenir la compatibilité avec Amazon Linux et les autres distributions majeures de Linux.

`mcrypt`

La `mcrypt` bibliothèque et l'PHPextension associée étaient obsolètes dans AL2 et ne sont plus présentes dans AL2023.

Upstream a rendu l'`mcrypt`extension PHP obsolète dans la PHP version 7.1, qui a été publiée pour la première fois en décembre 2016 et dont la version finale a été publiée en octobre 2019.

La `mcrypt` bibliothèque en amont a été publiée pour la dernière fois en 2007 et n'a pas effectué la migration depuis le contrôle des `cvs` révisions SourceForge requise pour les nouveaux commits en 2017, le commit le plus récent (et seulement pour les 3 années précédentes) datant de 2011, supprimant la mention du projet ayant un mainteneur.

Il est conseillé à tous `mcrypt` les utilisateurs restants de transférer leur code versOpenSSL, car il ne `mcrypt` sera pas ajouté à AL2023.

OpenJDK (7) **`java-1.7.0-openjdk`**

> **Note**
>
> AL2023 fournit plusieurs versions d'Amazon Corretto pour prendre en charge les charges de travail basées. Java Les packages OpenJDK 7 sont obsolètes dans AL2 et ne sont plus présents dans AL2023. Le plus ancien JDK disponible en AL2023 est fourni par Corretto 8.

Pour plus d'informations sur Java sur Amazon Linux, consultezJava dans AL2023.

Python 2.7

> **Note**
>
> AL2023 a supprimé Python 2.7, de sorte que tous les composants du système d'exploitation nécessitant Python sont écrits pour fonctionner avec Python 3. Pour continuer à utiliser une version de Python fournie et prise en charge par Amazon Linux, convertissez le code Python 2 en code pour Python 3.

Pour plus d'informations sur Python sur Amazon Linux, consultezPython dans AL2023.

`rsyslog-openssl`remplace **`rsyslog-gnutls`**

Le `rsyslog-gnutls` package est obsolète dans AL2 et n'est plus présent dans AL2023. Le `rsyslog-openssl` colis doit être un remplacement direct pour toute utilisation du `rsyslog-gnutls` package.

Service d'information réseau (NIS)/**yp**

Le Network Information Service (NIS), initialement appelé Pages Jaunes ou obsolète dans AL2, n'YPest plus présent dans AL2023. Cela inclut les packages suivants : `ypbind`, `ypserv`, `etyp-tools`. Cette fonctionnalité a NIS été supprimée dans AL2023 pour les autres packages qui s'intègrent à.

Obsolète dans AL2023

Cette section décrit les fonctionnalités qui existent dans AL2023 et qui seront probablement supprimées dans une future version d'Amazon Linux. Chaque section décrit en quoi consiste cette fonctionnalité et à quel moment elle devrait être supprimée d'Amazon Linux.

> **Note**
>
> Cette section sera mise à jour au fil du temps au fur et à mesure que l'écosystème Linux évolue et que les futures versions majeures d'Amazon Linux sont sur le point d'être publiées.

Rubriques

Support d'exécution 32 bits x86 (i686)

AL2023 conserve la capacité d'exécuter des binaires x86 (i686) 32 bits. Il est probable que la prochaine version majeure d'Amazon Linux ne prendra plus en charge l'exécution de fichiers binaires d'espace utilisateur 32 bits.

Berkeley DB () `libdb`

AL2023 est fourni avec la version 5.3.28 de la bibliothèque Berkeley DB (). `libdb` Il s'agit de la dernière version de Berkeley DB avant que la licence ne soit remplacée par la licence GNU Affero GPLv3 (AGPL), après la licence moins restrictive Sleepycat.

Dans AL2023, peu de packages dépendent encore de Berkeley DB (`libdb`), et la bibliothèque sera supprimée dans la prochaine version majeure d'Amazon Linux.

> **Note**
>
> Le gestionnaire de `dnf` packages d'AL2023 conserve le support en lecture seule pour une base de données au format Berkeley DB (BDB). `rpm` Ce support sera supprimé dans la prochaine version majeure d'Amazon Linux.

`cron`

Le package `cronie` était installé par défaut sur l'AMI AL2 et assurait la prise en charge de la méthode `crontab` traditionnelle de planification de tâches périodiques. Dans AL2023, n'`cronie`est pas inclus par défaut. Par conséquent, le support pour n'`crontab`est plus fourni par défaut.

Dans AL2023, vous pouvez éventuellement installer le `cronie` package pour utiliser des `cron` tâches classiques. Nous vous recommandons de migrer vers les temporisateurs `systemd` en raison des fonctionnalités supplémentaires fournies par `systemd`.

Il est possible qu'une future version d'Amazon Linux, probablement la prochaine version majeure, ne prenne plus en charge les `cron` tâches classiques et complète la transition vers `systemd` les minuteries. Nous vous recommandons de ne plus utiliser`cron`.

IMDSv1

Par défaut, les AMI AL2023 sont configurées pour se lancer en mode IMDSv2 -only, ce qui désactive l'utilisation de. IMDSv1 Il est toujours possible d'utiliser AL2023 avec IMDSv1 activé. Il est probable qu'une future version d'Amazon Linux n'appliquera que l'option IMDSv2 -only.

Pour plus d'informations sur la configuration de l'IMDS pour les AMI, consultez la section Configurer l'AMI dans le guide de l'utilisateur Amazon EC2.

`pcre`version 1

L'ancien `pcre` package est obsolète et sera supprimé dans la prochaine version majeure d'Amazon Linux. Le package `pcre2` est son successeur. Bien que les premières versions d'AL2023 aient été livrées avec un nombre limité de packages intégrés`pcre`, ces packages seront migrés vers `pcre2` AL2023. La `pcre` bibliothèque obsolète restera disponible dans AL2023.

> **Note**
>
> La version obsolète de ne `pcre` recevra pas de mises à jour de sécurité pendant toute la durée de vie d'AL2023. Pour plus d'informations sur le cycle de vie du `pcre` support et la durée pendant laquelle le package recevra les mises à jour de sécurité, consultez les déclarations de support du package figurant sur le `pcre` package.

System V init (`sysvinit`)

Bien qu'AL2023 conserve la rétrocompatibilité avec les scripts System V service (init), le `systemd` projet en amont, dans le cadre de sa version 254, a annoncé la suppression de la prise en charge des scripts de service System V et indiqué que le support serait supprimé dans une future version de. `systemd` Pour plus d'informations, consultez systemd.

L'AL2023 conservera la rétrocompatibilité avec les scripts System V service (init), mais les utilisateurs sont invités à passer à l'utilisation de fichiers `systemd` unités natifs afin de se préparer à la suppression de la prise en charge des scripts System V service (init) d'Amazon Linux, probablement dans la prochaine version majeure.

Comparaison entre AL2 et AL2023

Les rubriques suivantes décrivent les principales différences entre AL2 et AL2023.

Rubriques

- lsb_release et le package system-lsb-core
- Modifications du noyau AL2023 par rapport au noyau AL2
- Comparaison des packages installés sur les AMI Amazon Linux 2 et Amazon Linux 2023
- Comparaison des packages installés sur les AMI minimales Amazon Linux 2 et Amazon Linux 2023
- Comparaison des packages installés sur les images de conteneurs de base Amazon Linux 2 et Amazon Linux 2023

Packages ajoutés, mis à niveau et supprimés

AL2023 contient des milliers de packages logiciels utilisables. Pour obtenir la liste complète de tous les packages ajoutés, mis à niveau et supprimés dans AL2023 par rapport aux versions antérieures d'Amazon Linux, consultez Modifications des packages dans AL2023.

Pour demander qu'un package soit ajouté ou modifié dans AL2023, signalez un problème dans le référentiel amazon-linux-2023 sur. GitHub

Support pour chaque version

Pour AL2023, nous offrons cinq ans de support.

Pour plus d'informations, consultez Cadence de publication.

Modifications de dénomination et de gestion des versions

AL2023 prend en charge les mêmes mécanismes que ceux pris en charge par AL2 pour l'identification des plateformes. AL2023 introduit également de nouveaux fichiers pour l'identification des plateformes.

Pour plus d'informations, consultez Dénomination et gestion des versions.

Optimisations

AL2023 optimise le temps de démarrage afin de réduire le délai entre le lancement d'une instance et l'exécution de la charge de travail client. Ces optimisations concernent la configuration du noyau de l'instance Amazon EC2, les configurations `cloud-init` et les fonctionnalités intégrées aux packages du système d'exploitation telles que `kmod` et `systemd`.

Pour plus d'informations sur les optimisations, consultez Optimisations des performances et des opérations.

Python 2.7 a été remplacé par Python 3

AL2 fournit une assistance et des correctifs de sécurité pour Python 2.7 jusqu'en juin 2025, dans le cadre de notre engagement pour un support à long terme (LTS) pour les packages de base d'AL2. Cette prise en charge s'étend au-delà de la déclaration de la communauté Python 2.7 end-of-life de janvier 2020 en amont.

AL2 utilise le gestionnaire de `yum` paquets, qui dépend fortement de Python 2.7. Dans AL2023, le gestionnaire de packages `dnf` a migré vers Python 3 et ne nécessite plus Python 2.7. AL2023 a terminé sa transition vers Python 3.

> **Note**
>
> AL2023 a supprimé Python 2.7, de sorte que tous les composants du système d'exploitation nécessitant Python sont écrits pour fonctionner avec Python 3. Pour continuer à utiliser une version de Python fournie et prise en charge par Amazon Linux, convertissez le code Python 2 en code pour Python 3.

Pour plus d'informations relatives à Python sur Amazon Linux, consultez Python dans AL2023.

Mises à jour de sécurité

SELinux

Par défaut, Security Enhanced Linux (SELinux) pour AL2023 est `enabled` et défini sur le mode `permissive`. En mode `permissive`, les refus d'autorisation sont journalisés mais ne sont pas appliqués.

SELinux est une fonctionnalité de sécurité du noyau Amazon Linux, qui était `disabled` dans AL2. SELinux est un ensemble de fonctionnalités et d'utilitaires du noyau qui fournit une architecture de contrôle d'accès obligatoire (MAC) aux principaux sous-systèmes du noyau.

Pour plus d'informations, consultez Configuration des modes SELinux pour AL2023.

Pour plus d'informations sur les référentiels, les outils et les politiques SELinux, consultez Bloc-notes SELinux, Types de politique SELinux et Projet SELinux (langue française non garantie).

OpenSSL 3

AL2023 inclut la boîte à outils de cryptographie Open Secure Sockets Layer version 3 (OpenSSL 3). AL2023 prend en charge les protocoles réseau TLS 1.3 et TLS 1.2.

Par défaut, AL2 est fourni avec OpenSSL 1.0.2. Vous pouvez générer des applications avec OpenSSL 1.1.1.

Pour plus d'informations sur OpenSSL, consultez le guide de migration OpenSSL (langue française non garantie).

Pour plus d'informations sur la sécurité, consultez Mises à jour et fonctionnalités de sécurité.

IMDSv2

Par défaut, toutes les instances lancées avec l'AMI AL2023 nécessitent IMDSv2 -only et votre limite de sauts par défaut sera fixée à 2 pour permettre la prise en charge de la charge de travail conteneurisée. Cela se fait en définissant le paramètre `imds-support` sur `v2.0`. Pour plus d'informations, consultez Configurer l'AMI dans le guide de l'utilisateur Amazon EC2.

> **Note**
>
> La durée de validité du jeton de session peut avoir une valeur quelconque entre 1 seconde et 6 heures. Les adresses vers lesquelles diriger les demandes API pour les requêtes IMDSv2 sont les suivantes :
>
> - IPv4 : 169.254.169.254
> - IPv6 : fd00:ec2::254

Vous pouvez remplacer manuellement ces paramètres et les activer à IMDSv1 l'aide des propriétés de lancement de l'option Instance Metadata. Vous pouvez également utiliser les contrôles IAM pour appliquer différents IMDS paramètres. Pour plus d'informations sur la configuration et l'utilisation du service de métadonnées d'instance, consultez Utiliser IMDSv2, configurer les options de métadonnées d'instance pour les nouvelles instances et Modifier les options de métadonnées d'instance pour les instances existantes, dans le guide de l'utilisateur Amazon EC2.

Suppression de la mise à jour corrective à chaud log4j (**`log4j-cve-2021-44228-hotpatch`**)

> **Note**
>
> AL2023 n'est pas accompagné du package `log4j-cve-2021-44228-hotpatch`.

En réponse à CVE-2021-44228, Amazon Linux a publié une version empaquetée RPM du Hotpatch pour Apache Log4j pour AL1 et AL2. Dans l'annonce de l'ajout du correctif à chaud pour Amazon Linux, nous avons noté que « l'installation de la mise à jour corrective à chaud ne remplace pas la mise à jour vers une version log4j qui atténue les CVE-2021-44228 ou les CVE-2021-45046 ».

La mise à jour corrective à chaud était une mesure d'atténuation permettant de laisser le temps nécessaire au correctif `log4j`. La première version de disponibilité générale (GA) d'AL2023 a été publiée 15 mois après les CVE-2021-44228. Par conséquent, AL2023 n'est pas livré avec le correctif à chaud (activé ou non).

Les utilisateurs qui exécutent leurs propres `log4j` versions sur Amazon Linux doivent s'assurer qu'ils ont mis à jour les versions non affectées par CVE-2021-44228 ou CVE-2021-45046.

AL2023 fournit des conseils sur Mise à jour d'AL2023 pour que vous puissiez rester à jour des correctifs de sécurité. Les avis de sécurité sont publiés dans le Centre de sécurité Amazon Linux.

Mises à niveau déterministes pour la stabilité

Grâce à la fonctionnalité de mise à niveau déterministe via des référentiels versionnés, chaque AMI AL2023 est verrouillée par défaut sur une version de référentiel spécifique. Vous pouvez utiliser des mises à niveau déterministes pour améliorer la cohérence entre les versions et les mises à jour des packages. Chaque version, majeure ou mineure, inclut une version de référentiel spécifique.

Nouvelle dans AL2023, la mise à niveau déterministe est activée par défaut. Il s'agit d'une amélioration par rapport à la méthode de verrouillage manuelle et incrémentielle utilisée dans AL2 et dans d'autres versions antérieures.

Pour plus d'informations, consultez Utilisation de mises à niveau déterministes via un référentiel versionné sur AL2023.

Origine dans plusieurs sources en amont

AL2023 est basé sur RPM et inclut des composants provenant de plusieurs versions de Fedora et d'autres distributions, telles que CentOS 9 Stream. Le noyau Amazon Linux provient des versions de support à long terme (LTS) provenant directement de kernel.org et choisies indépendamment des autres distributions.

Pour plus d'informations, consultez Relation avec Fedora.

Système de fichiers racine d'AMI et type de volume Amazon EBS par défaut

L'AMI AL2023 et AL2 utilisent tous deux le système de fichiers XFS sur le système de fichiers racine. Pour AL2023, les options `mkfs` du système de fichiers racine de l'appareil sont encore optimisées pour Amazon EC2. AL2023 prend également en charge un certain nombre d'autres systèmes de fichiers que vous pouvez utiliser sur d'autres volumes pour répondre à vos besoins spécifiques.

Les AMI AL2023 utilisent les volumes `gp3` Amazon EBS par défaut, tandis que les AMI AL2 utilisent les volumes `gp2` Amazon EBS par défaut. Vous pouvez modifier le type de volume lorsque vous lancez une instance.

Pour plus d'informations sur les types de volume Amazon EBS, consultez Volumes à usage général Amazon EBS.

Pour plus d'informations sur le lancement d'une instance Amazon EC2, consultez Lancer une instance dans le guide de l'utilisateur Amazon EC2.

Service système de mise en réseau

Le service système `systemd-networkd` gère les interfaces réseau dans AL2023. Il s'agit d'une modification par rapport à AL2, qui utilise ISC dhclient ou `dhclient`.

Pour plus d'informations, consultez Service de mise en réseau.

Hiérarchie des groupes de contrôle unifiés (cgroup v2)

Un groupe de contrôle (cgroup) est une fonctionnalité du noyau Linux permettant d'organiser hiérarchiquement les processus et de répartir les ressources système entre eux. Les groupes

de contrôle sont très fréquemment utilisés pour implémenter un environnement d'exécution de conteneur, et par `systemd`.

Les supports AL2 et AL2023 cgroupv1 les prennent en charge. cgroupv2 Cela est notable si vous exécutez des charges de travail conteneurisées, par exemple lors de l'Utilisation d'AMI Amazon ECS basées sur AL2023 pour héberger des charges de travail conteneurisées.

Bien que l'AL2023 contienne toujours du code permettant au système de fonctionner en utilisantcgroupv1, cette configuration n'est ni recommandée ni prise en charge, et elle sera complètement supprimée dans une future version majeure d'Amazon Linux.

Il existe une documentation abondante sur les interfaces du noyau Linux de bas niveau, ainsi qu'une documentation sur la délégation des groupes de contrôle systemd (langue française non garantie).

Un cas d'utilisation courant en dehors des conteneurs consiste à créer des `systemd` unités dont les ressources système peuvent être limitées. Pour plus d'informations, consultez systemd.resource-control.

Planification de tâches

Le package `cronie` était installé par défaut sur l'AMI AL2 et assurait la prise en charge de la méthode `crontab` traditionnelle de planification de tâches périodiques. Dans AL2023, n'`cronie`est pas inclus par défaut. Par conséquent, le support pour n'`crontab`est plus fourni par défaut.

Vous pouvez éventuellement installer le package `cronie` pour utiliser des tâches `cron` classiques. Nous vous recommandons de migrer vers les temporisateurs `systemd` en raison des fonctionnalités supplémentaires fournies par `systemd`.

Packages pour glibc, gcc et binutils

AL2023 inclut un grand nombre des packages de base d'AL2.

Nous avons mis à jour les trois packages de chaînes d'outils de base suivants pour AL2023.

Nom du package	AL2	AL2023
glibc	2,26	2,34
gcc	7.3	11,3

Nom du package	AL2	AL2023
binutils	2,29	2,39

Pour plus d'informations, consultez Packages de chaîne d'outils de base glibc, gcc, binutils.

Gestionnaire de packages

Dans AL2023, l'outil de gestion des packages logiciels par défaut est DNF. DNF est le successeur de YUM, l'outil de gestion de packages dans AL2.

Pour plus d'informations, consultez Outil de gestion de package.

Système de journalisation

Dans AL2023, le package du système de journalisation a changé par rapport à AL2. AL2023 n'installe pas `rsyslog` par défaut, de sorte que les fichiers journaux texte tels que `/var/log/messages` qui étaient disponibles dans AL2 ne sont pas disponibles par défaut. La configuration par défaut pour AL2023 est `systemd-journal`, qui peut être examinée à l'aide de `journalctl`. `rsyslog` est un package facultatif dans AL2023, mais nous recommandons la nouvelle interface `journalctl` basée sur `systemd` et les packages associés. Pour plus d'informations, consultez la page de documentation `journalctl`.

Modifications des packages pour **`curl`** et **`libcurl`**

AL2023 sépare les protocoles et fonctionnalités courants des packages `curl` et `libcurl` dans `curl-minimal` et `libcurl-minimal`. Cela réduit l'empreinte sur le disque, la mémoire et les dépendances pour la plupart des utilisateurs, et constitue le package par défaut pour les AMI et les conteneurs AL2023.

Si toutes les fonctionnalités de `curl` sont requises, par exemple pour la prise en charge de `gopher://`, exécutez les commandes suivantes pour installer les packages `curl-full` et `libcurl-full`.

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

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

GNU Privacy Guard (GNUPG)

AL2023 sépare les fonctionnalités minimales et complètes du package `gnupg2` dans les packages `gnupg2-minimal` et `gnupg2-full`. Par défaut, seul le package `gnupg2-minimal` est installé. Cela fournit les fonctionnalités minimales requises pour vérifier les signatures numériques sur les packages `rpm`.

Pour bénéficier de fonctionnalités supplémentaires `gnupg2`, telles que la possibilité de télécharger des clés à partir d'un serveur de clés, assurez-vous que le package `gnupg2-full` est installé. Exécutez la commande suivante pour remplacer `gnupg2-minimal` par `gnupg2-full`.

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

Amazon Corretto en tant que machine virtuelle Java par défaut

AL2023 est livré avec Amazon Corretto comme kit de développement Java (JDK) par défaut (et unique). Tous les packages Java basés dans AL2023 sont tous construits avecAmazon Corretto 17.

Si vous migrez depuis AL2, vous pouvez passer en douceur de la `OpenJDK` version équivalente sur AL2 à. Amazon Corretto

AWS CLI v2

L'AL2023 est livré avec AWS CLI la version 2, tandis que l'AL2 est livré avec la version 1 du. AWS CLI

UEFI préférée

Par défaut, toutes les instances lancées avec l'AMI AL2023 sur des types d'instances prenant en charge le microprogramme UEFI sont lancées en mode UEFI. Cela est dû à la définition du paramètre Mode de démarrage de l'AMI sur `uefi-preferred`. Pour plus d'informations, consultez la section Modes de démarrage dans le guide de l'utilisateur Amazon EC2.

Modifications de la configuration par défaut du serveur SSH

Pour l'AMI AL2023, nous avons modifié les types de clés d'hôte `sshd` générées avec la version. Nous avons également supprimé certains types de clés existants pour éviter de les générer au moment du lancement. Les clients doivent prendre en charge les protocoles `rsa-sha2-256` et `rsa-`

`sha2-512`, ou `ssh-ed25519` en utilisant une clé `ed25519`. Par défaut, les signatures `ssh-rsa` sont désactivées.

En outre, les paramètres de configuration AL2023 figurant dans le fichier `sshd_config` par défaut contiennent `UseDNS=no`. Ce nouveau paramètre signifie que les déficiences DNS sont moins susceptibles de vous empêcher d'établir des sessions `ssh` avec vos instances. L'inconvénient est que les entrées de ligne `from=`*`hostname.domain`*`,`*`hostname.domain`* figurant dans vos fichiers `authorized_keys` ne sont pas résolues. Comme `sshd` ne tente plus de résoudre les noms DNS, chaque valeur *`hostname.domain`* séparée par des virgules doit être traduite en une IP address correspondante.

Pour plus d'informations, consultez Configuration du serveur SSH par défaut.

Extra Packages for Enterprise Linux (EPEL)

Extra Packages for Enterprise Linux (EPEL) est un projet de la communauté Fedora dont l'objectif est de créer un grand ensemble de packages pour les systèmes d'exploitation Linux d'entreprise. Ce projet a généré principalement les packages RHEL et CentOS. AL2 offre un haut niveau de compatibilité avec CentOS 7. Par conséquent, de nombreux packages EPEL7 fonctionnent sur AL2. Toutefois, AL2023 ne prend pas en charge les référentiels EPEL ou de type EPEL.

Utiliser cloud-init

Dans AL2023, cloud-init gère le référentiel de packages. Par défaut, dans les versions antérieures d'Amazon Linux, cloud-init installait les mises à jour de sécurité. Ce n'est pas le cas par défaut pour AL2023. Les nouvelles fonctionnalités de mise à niveau déterministe pour la mise à jour de `releasever` au lancement décrivent la façon dont AL2023 active les mises à jour des packages au lancement. Pour plus d'informations, consultez Gérez les mises à jour des packages et du système d'exploitation dans AL2023 et Mises à niveau déterministes pour la stabilité.

Avec AL2023, vous pouvez utiliser cloud-init avec SELinux. Pour plus d'informations, consultez Utilisation de cloud-init pour activer le mode `enforcing`.

Cloud-init charge le contenu de configuration avec cloud-init à partir d'emplacements distants via HTTP(S). Dans les versions antérieures, Amazon Linux ne vous avertit pas quand des ressources distantes ne sont pas disponibles. Dans AL2023, l'indisponibilité des ressources distantes génère une erreur fatale et fait échouer l'exécution de cloud-init. Ce changement de comportement par rapport à AL2 fournit un comportement par défaut de « fermeture en cas d'échec » plus sûr.

Pour plus d'informations, consultez Version cloud-init personnalisée et la documentation cloud-init (langue française non garantie).

Prise en charge d'un bureau graphique

AL2023 est centré sur le cloud et optimisé pour l'utilisation d'Amazon EC2 et n'inclut actuellement aucun environnement graphique ou de bureau. Pour nous faire part de vos commentaires GitHub, consultez https://github.com/.

Triplet de compilateur

AL2023 définit le triplet de compilateur pour GCC et `LLVM` afin d'indiquer qu'`amazon` est le fournisseur.

Ainsi, l'élément `aarch64-redhat-linux-gcc` d'AL2 devient `aarch64-amazon-linux-gcc` sur AL2023.

Cela devrait être totalement transparent pour la plupart des utilisateurs et pourrait n'affecter que ceux qui construisent des compilateurs sur AL2023.

Packages x86 (i686) 32 bits

Dans le cadre de la version 2014.09 d'AL1, il a été annoncé que ce serait la dernière version à produire des AMI 32 bits. Ainsi, depuis la version 2015.03 d'AL1, Amazon Linux ne prend plus en charge l'exécution du système en mode 32 bits. AL2 offrait une prise en charge limitée de l'environnement d'exécution pour les fichiers binaires 32 bits sur des hôtes x86-64 et ne fournissait pas de packages de développement permettant de créer de nouveaux fichiers binaires 32 bits. AL2023 n'inclut plus aucun package d'espace utilisateur 32 bits. Nous vous recommandons de terminer votre transition vers le code 64 bits.

Si vous devez exécuter des fichiers binaires 32 bits sur AL2023, il est possible d'utiliser l'espace utilisateur 32 bits d'AL2 au sein d'un conteneur AL2 exécuté par-dessus AL2023.

`lsb_release` et le package `system-lsb-core`

Historiquement, certains logiciels invoquaient la commande `lsb_release` (fournie dans AL2 par le package `system-lsb-core`) pour obtenir des informations sur la distribution Linux sur laquelle ils

s'exécutaient. Le projet Linux Standards Base (LSB) a introduit cette commande et les distributions Linux l'ont adoptée. Les distributions Linux ont évolué pour utiliser le standard simplifié consistant à conserver ces informations dans `/etc/os-release` et d'autres fichiers connexes.

Le standard `os-release` est issu de `systemd`. Pour plus d'informations, consultez la documentation systemd sur os-release (langue française non garantie).

AL2023 n'est pas fourni avec la commande `lsb_release` et n'inclut pas le package `system-lsb-core`. Le logiciel doit terminer la transition vers le standard `os-release` pour maintenir la compatibilité avec Amazon Linux et les autres distributions majeures de Linux.

Modifications du noyau AL2023 par rapport au noyau AL2

AL2023 apporte le noyau 6.1, ainsi que de nombreuses modifications de configuration afin d'optimiser davantage Amazon Linux pour le cloud. Pour la plupart des utilisateurs, ces modifications doivent être totalement transparentes.

Modifications de configuration du noyau axées sur la sécurité

Option **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
CONFIG_BUG_ON_DATA_CORRUPTION	n	y	n	y	y	y
CONFIG_DEFAULT_MMAP_MIN_ADDR	4096	4096	4096	4096	65536	65536
CONFIG_DEVMEM	n	y	n	y	n	n
CONFIG_DEVPORT	n	y	n	y	n	n

Option **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
CONFIG_FORTIFY_SOURCE	n	y	n	y	y	y
CONFIG_HARDENED_USERCOPY_FALLBACK	N/A	N/A	y	y	N/A	N/A
CONFIG_INIT_ON_ALLOC_DEFAULT_ON	N/A	N/A	n	n	n	n
CONFIG_INIT_ON_FREE_DEFAULT_ON	N/A	N/A	n	n	n	n
CONFIG_IOMMU_DEFAULT_DMA_STRICT	N/A	N/A	N/A	N/A	n	n
CONFIG_LDISC_AUTOLOAD	y	y	y	y	n	n
CONFIG_SCHED_CORE	N/A	N/A	N/A	N/A	N/A	y
CONFIG_SCHED_STACK_END_CHECK	n	y	n	y	y	y

Option **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
CONFIG_SECURITY_DMESG_RESTRICT	n	n	n	n	y	y
CONFIG_SECURITY_SELINUX_DISABLE	y	y	y	y	n	n
CONFIG_SHUFFLE_PAGE_ALLOCATOR	N/A	N/A	y	y	y	y
CONFIG_SLAB_FREELIST_HARDENED	n	y	y	y	y	y
CONFIG_SLAB_FREELIST_RANDOM	n	n	y	y	y	y

Modifications de configuration du noyau axées sur la sécurité spécifiques à x86-64

Option **CONFIG**	AL2/4.14/x86_64	AL2/5.10/x86_64	AL2023/6.1/x86_64
CONFIG_AMD_IOMMU	y	y	y
CONFIG_AMD_IOMMU_V2	m	m	y

Option **CONFIG**	AL2/4.14/x86_64	AL2/5.10/x86_64	AL2023/6.1/x86_64
CONFIG_RANDOMIZE_MEMORY	N/A	y	y

Modifications de configuration du noyau axées sur la sécurité spécifiques à aarch64 (ARM/Graviton)

Option **CONFIG**	AL2/4.14/aarch64	AL2/5.10/aarch64	AL2023/6.1/aarch64
CONFIG_ARM64_PTR_AUTH	N/A	y	y
CONFIG_ARM64_PTR_AUTH_KERNEL	N/A	N/A	y
CONFIG_ARM64_SW_TTBR0_PAN	y	y	y

/dev/mem, **/dev/kmem** et **/dev/port**

Amazon Linux 2023 désactive `/dev/mem`, et `/dev/port` (`CONFIG_DEVMEMetCONFIG_DEVPORT`) complètement, en s'appuyant sur les restrictions déjà en place dans AL2.

Le `/dev/kmem` code a été complètement supprimé de Linux dans le noyau 5.13, et bien qu'il ait été désactivé dans AL2, il n'est désormais plus applicable à AL2023.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

FORTIFY_SOURCE

AL2023 est activé `CONFIG_FORTIFY_SOURCE` sur toutes les architectures prises en charge. Il s'agit d'une fonctionnalité de renforcement de la sécurité. Lorsque le compilateur peut déterminer et valider les tailles de mémoire tampon, cette fonctionnalité permet de détecter les dépassements de mémoire tampon dans les fonctions de chaîne et de mémoire courantes.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Chargement automatique de Line Discipline () **CONFIG_LDISC_AUTOLOAD**

Le noyau AL2023 ne chargera pas automatiquement les disciplines de ligne, par exemple par un logiciel utilisant le `TIOCSETDioctl`, sauf si la demande provient d'un processus disposant des `CAP_SYS_MODULE` autorisations.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

dmesgaccès pour les utilisateurs non privilégiés () **CONFIG_SECURITY_DMESG_RESTRICT**

Par défaut, AL2023 n'autorise pas l'accès des utilisateurs non privilégiés à. `dmesg`

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Désactiver SELinux **selinuxfs**

AL2023 désactive l'option obsolète du `CONFIG_SECURITY_SELINUX_DISABLE` noyau, qui activait une méthode d'exécution pour désactiver SELinux avant le chargement de la politique.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Autres modifications de configuration du noyau

Option **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
CONFIG_HZ	100	250	100	250	100	100
CONFIG_NR_CPUS	4096	8192	4096	8192	512	512
CONFIG_PANIC_ON_OOPS	y	n	y	n	y	y

Option **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
CONFIG_PANIC_ON_OOPS_VALUE	1	0	1	0	1	1
CONFIG_PPP	m	m	m	m	n	n
CONFIG_SLIP	m	m	m	m	n	n
CONFIG_XEN_PV	N/A	y	N/A	n	N/A	n

CONFIG_HZ

L'AL2023 passe `CONFIG_HZ` à 100 sur `x86-64` les deux `aarch64` plateformes.

CONFIG_NR_CPUS

AL2023 définit `CONFIG_NR_CPUS` un nombre plus proche du nombre maximal de cœurs de processeur trouvés dans Amazon EC2.

Panique sur OOPS

Le noyau AL2023 paniquera lorsqu'il s'opposera. Cette fonctionnalité est équivalente au démarrage avec `oops=panic` depuis la ligne de commande du noyau.

Un oops du noyau correspond à la détection par le noyau d'une erreur interne susceptible d'affecter la fiabilité ultérieure du système.

Prise en charge de PPP et SLIP

AL2023 ne prend pas en charge les protocoles PPP ou SLIP.

Prise en charge des invités PV sur Xen

L'AL2023 ne prend pas en charge l'exécution en tant qu'invité Xen PV.

Prise en charge des systèmes de fichiers par le noyau

Plusieurs modifications ont été apportées aux systèmes de fichiers que le noyau d'AL2 permettra de monter, ainsi que des modifications dans les schémas de partitionnement que le noyau analysera.

Option **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
CONFIG_AFS_FS	n	m	n	m	n	n
CONFIG_AF_RXRPC	n	m	n	m	n	n
CONFIG_BSD_DISKLABEL	y	y	y	y	n	n
CONFIG_CRAMFS	m	m	m	m	n	n
CONFIG_CRAMFS_BLOCKDEV	N/A	N/A	y	n	N/A	N/A
CONFIG_DM_CLONE	N/A	N/A	n	n	n	n
CONFIG_DM_ERA	m	n	m	n	n	n
CONFIG_DM_INTEGRITY	n	m	n	m	m	m
CONFIG_DM_LOG_WRITES	n	n	m	m	m	m

Option **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
CONFIG_DM_SWITCH	m	n	m	n	n	n
CONFIG_DM_VERITY	m	n	m	n	n	n
CONFIG_ECRYPT_FS	n	m	n	m	n	n
CONFIG_EXFAT_FS	N/A	N/A	m	m	m	m
CONFIG_EXT2_FS	n	m	n	m	n	n
CONFIG_EXT3_FS	n	m	n	m	n	n
CONFIG_GFS2_FS	m	m	m	m	n	n
CONFIG_HFSPLUS_FS	n	m	n	m	n	n
CONFIG_HFS_FS	n	m	n	m	n	n
CONFIG_JFS_FS	n	m	n	m	n	n
CONFIG_LDM_PARTITION	n	y	n	y	n	n

Option **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
CONFIG_MAC_PARTITION	n	y	n	y	n	n
CONFIG_NFS_V2	n	m	n	m	n	n
CONFIG_NTFS_FS	n	m	n	n	n	n
CONFIG_ROMFS_FS	n	m	n	m	n	n
CONFIG_SOLARIS_X86_PARTITION	n	y	n	y	n	n
CONFIG_SQUASHFS_ZSTD	n	y	n	y	y	y
CONFIG_SUN_PARTITION	n	y	n	y	n	n

Prise en charge du système de fichiers Andrew (AFS)

Le noyau n'intègre plus la prise en charge du système de fichiers `afs`. AL2 n'était pas livré avec le support de l'espace utilisateur pour. `afs`

Prise en charge de cramfs

Le noyau n'intègre plus la prise en charge du système de fichiers `cramfs`. Le successeur d'AL2023 est le système de `squashfs` fichiers.

Prise en charge des étiquettes de disque BSD

Le noyau n'intègre plus la prise en charge des étiquettes de disque BSD. Si la lecture de volumes avec des étiquettes de disque BSD est requise, différents systèmes BSD peuvent être lancés.

Modifications apportées à Device Mapper

Plusieurs modifications ont été apportées aux cibles Device Mapper configurées dans le noyau AL2023.

eCryptFs soutien

Le système de fichiers `ecryptfs` est devenu obsolète dans Amazon Linux. Les composants de l'espace utilisateur de `ecryptfs` étaient présents dans AL1, supprimés dans AL2, et AL2023 ne construit plus le noyau avec support. `ecryptfs`

exFAT

Support du système de `exFAT` fichiers a été ajouté dans le noyau 5.10 dans AL2. Il n'était pas présent lors du lancement d'AL2 avec un noyau 4.14. AL2023 continue de prendre en charge le système de `exFAT` fichiers.

Systèmes de fichiers ext2, ext3 et ext4

L'AL2023 est livré avec `CONFIG_EXT4_USE_FOR_EXT2` cette option, ce qui signifie que le code du système de `ext4` fichiers sera utilisé pour lire les anciens systèmes de `ext2` fichiers.

CONFIG_GFS2_FS

Le noyau n'est plus doté de CONFIG_GFS2_FS.

Prise en charge du système de fichiers Apple HFS étendu (HFS+)

Dans AL2, seuls les `x86-64` noyaux étaient créés avec le support du système de `hfsplus` fichiers. Le noyau AL2 5.15 ne prend en `hfsplus` charge aucune architecture. Dans AL2023, nous finalisons la dépréciation du support `hfsplus` dans Amazon Linux.

Prise en charge du système de fichiers HFS

Dans AL2, seuls les `x86-64` noyaux étaient créés avec le support du système de `hfs` fichiers. Le noyau AL2 5.15 ne prend en `hfs` charge aucune architecture. Dans AL2023, nous finalisons la dépréciation du support `hfs` dans Amazon Linux.

Prise en charge du système de fichiers JFS

Dans AL2, seuls les `x86-64` noyaux étaient créés avec le support du système de `jfs` fichiers. Le noyau AL2 5.15 ne prend en `jfs` charge aucune architecture. Ni AL1 ni AL2 ne sont fournis avec l'espace utilisateur JFS. Dans AL2023, nous finalisons la dépréciation du support `jfs` dans Amazon Linux.

Le noyau Linux en amont envisage de supprimer `JFS`. Par conséquent, si vous avez des données sur un système de `JFS` fichiers, vous devez les migrer vers un autre système de fichiers.

WindowsSupport du gestionnaire de disques logiques (disque dynamique) (**`CONFIG_LDM_PARTITION`**)

AL2023 ne prend plus en charge Windows 2000 Windows XP les disques Windows Vista dynamiques dotés de partitions de MS-DOS style. Ce code n'a jamais pris en charge les nouveaux disques dynamiques basés sur GPT introduits avecWindows Vista.

Prise en charge du mappage de partition Macintosh

AL2023 ne prend plus en charge la carte de partition classique du Macintosh. Les versions modernes de macOS créeront des tables de partitions GPT modernes par défaut sur cet ancien type.

Prise en charge de NFSv2

AL2023 ne prend plus en charge NFSv2, mais continue de prendre en charge NFSv3, NFSv4, NFSv4.1 et NFSv4.2. Nous vous recommandons de migrer vers NFSv3 ou une version ultérieure.

NTFS (**`CONFIG_NTFS_FS`**)

Le `ntfs3` code a été remplacé `ntfs` pour accéder aux systèmes de fichiers NTFS sur Amazon Linux à partir du noyau 5.10 dans AL2. AL2023 n'inclut plus le `ntfs` code et s'appuie exclusivement sur le `ntfs3` code pour accéder aux systèmes de fichiers NTFS.

Système de fichiers romfs

Le système de fichiers `squashfs` est le successeur du système de fichiers `romfs` d'Amazon Linux, et le noyau AL2023 n'est plus doté de la prise en charge de `romfs`.

Format de partition de disque dur Solaris x86

AL2023 ne prend plus en charge le format de partition de disque dur Solaris x86.

Compression **squashfs**zstd

AL2023 ajoute la prise en charge des systèmes de `squashfs` fichiers zstd compressés sur toutes les architectures prises en charge.

Prise en charge de la table de partition Sun

AL2023 ne prend plus en charge le format de table de partition Sun (`CONFIG_SUN_PARTITION`).

Comparaison des packages installés sur les AMI Amazon Linux 2 et Amazon Linux 2023

Comparaison des RPM présents sur les AMI standard Amazon Linux 2 et AL2023.

Package	TOUTES LES 2 AMI	TOUS LES 2023 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`		2.4.1
`amazon-linux-extras`	2.0.3	
`amazon-linux-extras-yum-plugin`	2.0.3	
`amazon-linux-repo-s3`		2023,4.20240513
`amazon-linux-sb-keys`		2023.1
`amazon-rpm-config`		228
`amazon-ssm-agent`	3.3.131.0	3,3.380,0
`at`	3,113	3,1,23

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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	9,16,48
bind-libs-lite	9,11.4	
bind-license	9,11.4	9,16,48
bind-utils	9,11.4	9,16,48
binutils	2,29.1	2,39
blktrace	1.0.5	
boost-date-time	1,53,0 (x86_64)	
boost-filesystem		1,75,0

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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	2023,2,64	2023,2,64
c-ares		1.19.0
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
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	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
crontabs	1.11	1.11
crypto-policies		20220428
crypto-policies-scripts		20220428
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	2,127
cyrus-sasl-plain	2.1.26	2,127
dbus	1,1,24	1,12,28
dbus-broker		32
dbus-common		1,12,28
dbus-libs	1,1,24	1,12,28
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	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
dosfstools	3,0,20	4.2
dracut	033	055
dracut-config-ec2	2.0	3.0
dracut-config-generic	033	055
dwz		0,14

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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 (aarch64)	
efi-filesystem		5
efi-srpm-macros		5
efivar		38
efivar-libs	31 (aarch64)	38
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

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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-langpack	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
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 (aarch64)	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
grub2-efi-aa64-ec2	2,06 (aarch64)	2,06 (aarch64)
grub2-efi-aa64-modules	2,06 (novembre)	
grub2-efi-x64-ec2	2,06 (x86_64)	2,06 (x86_64)
grub2-pc	2,06 (x86_64)	
grub2-pc-modules	2,06 (novembre)	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
hunspell-en-GB	0,20121024	0,20140811,1
hunspell-en-US	0,20121024	0,20140811,1
hunspell-filesystem		1.7.0
hwdata	0,252	0,353
info	5.1	6.7

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
inih		49
initscripts	9,49,47	10,09
iproute	5.10.0	5.10.0
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	
jitterentropy		3.4.1
jq		1.7.1
json-c	0,11	0,14
kbd	1,1,5	2.4.0
kbd-legacy	1,1,5	
kbd-misc	1,1,5	2.4.0
kernel	5,1,215	6,1,90
kernel-livepatch-repo-s3		2023,4.20240513
kernel-srpm-macros		1.0
kernel-tools	5,1,215	6,1,90
keyutils	1.5.8	1.6.3

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
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		3.5.3
libargon2		27/12/2017
libassuan	2.1.0	2.5.5
libattr	2,4,46	2.5.1
libbasicobjects	0,11	0,11
libblkid	2,30,2	2,37,4
libcap	2,54	2,48
libcap-ng	0,7,5	0.8.2

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
libdnf		0,69,0
libdrm	2,4,97	
libdwarf	20130207 (x86_64)	
libeconf		0,4,0
libedit	3.0	3.1
libestr	0,19	
libev		4,33

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
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	0,13	0,13

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
libpkgconf		1.8.0
libpng	1.5,13	
libpsl	0,21,5	0,21,1
libpwquality	1.2.3	1.4.4
libref_array	0,15	0,15
librepo		1.14,5
libreport-filesystem		2.15.2

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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	1,1,5	2.9.4
libsss_nss_idmap	1,1,5	2.9.4
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	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
libverto	0,2,5	0,3.2
libverto-libev		0,3.2
libverto-libevent	0,2,5	
libwebp	0.3.0	
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libxml2-python	2.9.1	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
libyaml	0,14	0,2,5
libzstd		1.5.5
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	
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)

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
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	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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	7,4 p1	8,7 p1
openssh-clients	7,4 p1	8,7 p1
openssh-server	7,4 p1	8,7 p1
openssl	1,02k	3,0.8
openssl-libs	1,02k	3,0.8
openssl-pkcs11		0,4,12
os-prober	1.58	1,77
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

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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		2021,0726
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	0.8.1	
pkgconf		1.8.0
pkgconfig	0,27,1	
pkgconf-m4		1.8.0

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
psmisc	22,20	23,4
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0,5.3	
pystache	0,5.3	
python	2.7,18	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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	0,19	
python2-rpm	4.11.3	
python2-rsa	3.4.1	
python2-s3transfer	0,3.3	
python2-setuptools	41,2,0	
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

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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		0,1,6
python3-oauthlib		3.0.2
python3-pip	20.2.2	
python3-pip-wheel		21.3.1
python3-ply		3,11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3,0,24
python3-pycparser		2,20
python3-pyrsistent		0,17.3
python3-pyserial		3.4

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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		0,12
python3-setools		4.4.1
python3-setuptools	49,13	59,6,0
python3-setuptools-wheel		59,6,0
python3-simplejson	3.2.0	
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	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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	1,016	
python-jinja2	2.7.2	
python-jsonpatch	1.2	
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	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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	37,22
selinux-policy-targeted	3.13.1	37,22
setserial	2,17	
setup	2,8,71	2.13.7
setuptool	1,19,11	
sgpio	1.2.0.10	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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	1,1,5	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	252,16
systemd-libs	219	252,16
systemd-networkd		252,16
systemd-pam		252,16
systemd-resolved		252,16
systemd-sysv	219	
systemd-udev		252,16

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
system-release	2	2023,4.20240513
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	6,24,07
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
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

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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
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	

Package	TOUTES LES 2 AMI	TOUS LES 2023 AMI
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

Comparaison des packages installés sur les AMI minimales Amazon Linux 2 et Amazon Linux 2023

Comparaison des RPM présents sur les AMI minimales Amazon Linux 2 et AL2023.

Package	AL2 Minimale	AL2023 Minimale
acl	2,2,51	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.4.1
amazon-linux-extras	2.0.3	
amazon-linux-repo-s3		2023,4.20240513
amazon-linux-sb-keys		2023.1
audit	2.8.1	3,0.6
audit-libs	2.8.1	3,0.6

Package	AL2 Minimale	AL2023 Minimale
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	2023,2,64	2023,2,64
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
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	

Package	AL2 Minimale	AL2023 Minimale
crontabs	1.11	
crypto-policies		20220428
cryptsetup-libs	1.7.4	2.6.1
curl	8.3.0	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.26	2,127
dbus	1,1,24	1,12,28
dbus-broker		32
dbus-common		1,12,28
dbus-libs	1,1,24	1,12,28
device-mapper	1,02,170	1,02,185
device-mapper-libs	1,02,170	1,02,185
dhclient	4.2.5	
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

Package	AL2 Minimale	AL2023 Minimale
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 (aarch64)	
efi-filesystem		5
efivar		38
efivar-libs	31 (aarch64)	38
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

Package	AL2 Minimale	AL2023 Minimale
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
glibc-all-langpacks	2,26	2,34
glibc-common	2,26	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

Package	AL2 Minimale	AL2023 Minimale
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 (aarch64)	
grub2-efi-aa64-ec2	2,06 (aarch64)	2,06 (aarch64)
grub2-efi-aa64-modules	2,06 (novembre)	
grub2-efi-x64-ec2	2,06 (x86_64)	2,06 (x86_64)
grub2-pc	2,06 (x86_64)	
grub2-pc-modules	2,06 (novembre)	2,06
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		0,353
info	5.1	

Package	AL2 Minimale	AL2023 Minimale
inih		49
initscripts	9,49,47	10,09
iproute	5.10.0	5.10.0
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
kbd-misc		2.4.0
kernel	4,14,343	6,1,90
kernel-livepatch-repo-s3		2023,4.20240513
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

Package	AL2 Minimale	AL2023 Minimale
less	458	608
libacl	2,2,51	2.3.1
libarchive		3.5.3
libargon2		27/12/2017
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
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

Package	AL2 Minimale	AL2023 Minimale
libedit	3.0	3.1
libestr	0,19	
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	
libidn	1,28	
libidn2	2.3.0	2.3.2
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libmetalink	0,13	
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2,30,2	2,37,4
libnetfilter_conntrack	1.0.6	

Package	AL2 Minimale	AL2023 Minimale
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
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

Package	AL2 Minimale	AL2023 Minimale
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		4.4.33
libxml2	2.9.1	2.10.4
libyaml	0,14	0,2,5
libzstd		1.5.5
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	

Package	AL2 Minimale	AL2023 Minimale
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
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

Package	AL2 Minimale	AL2023 Minimale
openldap	2,4,44	2,4,57
openssh	7,4 p1	8,7 p1
openssh-clients	7,4 p1	8,7 p1
openssh-server	7,4 p1	8,7 p1
openssl	1,02k	3,0.8
openssl-libs	1,02k	3,0.8
openssl-pkcs11		0,4,12
os-prober	1.58	1,77
p11-kit	0,23,22	0,24.1
p11-kit-trust	0,23,22	0,24.1
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	10,40
pcre2-syntax		10,40
pinentry	0.8.1	
pkgconfig	0,27,1	
policycoreutils	2,5	3.4

Package	AL2 Minimale	AL2023 Minimale
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	
python2-jsonschema	2.5.1	
python2-oauthlib	2.0.1	
python2-pyasn1	0,19	
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

Package	AL2 Minimale	AL2023 Minimale
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
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

Package	AL2 Minimale	AL2023 Minimale
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		0,1,6
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

Package	AL2 Minimale	AL2023 Minimale
python3-requests		2.25.1
python3-rpm		4.16.1.3
python3-ruamel-yaml		0,16.6
python3-ruamel-yaml-clib		0,12
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-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	

Package	AL2 Minimale	AL2023 Minimale
python-enum34	1.0.4	
python-idna	2,4	
python-iniparse	0.4	
python-ipaddress	1,016	
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	
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	

Package	AL2 Minimale	AL2023 Minimale
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
sed	4.2.2	4.8
selinux-policy	3.13.1	37,22
selinux-policy-targeted	3.13.1	37,22
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	

Package	AL2 Minimale	AL2023 Minimale
sqlite-libs		3,40,0
sudo	1,8,23	1,9,15
sysctl-defaults	1.0	1.0
systemd	219	252,16
systemd-libs	219	252,16
systemd-networkd		252,16
systemd-pam		252,16
systemd-resolved		252,16
systemd-sysv	219	
systemd-udev		252,16
system-release	2	2023,4.20240513
sysvinit-tools	2,88	
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

Package	AL2 Minimale	AL2023 Minimale
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
zram-generator-defaults		1.1.2
zstd		1.5.5

Comparaison des packages installés sur les images de conteneurs de base Amazon Linux 2 et Amazon Linux 2023

Comparaison des RPM présents sur les images des conteneurs de base Amazon Linux 2 et AL2023.

Package	Conteneur AL2	Conteneur AL2023
alternatives		1.15

Package	Conteneur AL2	Conteneur AL2023
amazon-linux-extras	2.0.3	
amazon-linux-repo-cdn		2023,4.20240513
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	2023,2,64	2023,2,64
chkconfig	1.7.4	
coreutils	8,22	
coreutils-single		8,32
cpio	2,12	
crypto-policies		20220428
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

Package	Conteneur AL2	Conteneur AL2023
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
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	

Package	Conteneur AL2	Conteneur AL2023
json-c		0,14
keyutils-libs	1.5.8	1.6.3
krb5-libs	1.15.1	1,21
libacl	2,2,51	2.3.1
libarchive		3.5.3
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
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

Package	Conteneur AL2	Conteneur AL2023
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	0,13	
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
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

Package	Conteneur AL2	Conteneur AL2023
libuuid	2,30,2	2,37,4
libverto	0,2,5	0,3.2
libxcrypt		4.4.33
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
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	

Package	Conteneur AL2	Conteneur AL2023
nss-tools	3,90,0	
nss-util	3,90,0	
openldap	2,4,44	
openssl-libs	1,02k	3,0.8
p11-kit	0,23,22	0,24.1
p11-kit-trust	0,23,22	0,24.1
pcre	8,32	
pcre2		10,40
pcre2-syntax		10,40
pinentry	0.8.1	
popt	1.13	1,18
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
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

Package	Conteneur AL2	Conteneur AL2023
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	
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	

Package	Conteneur AL2	Conteneur AL2023
sqlite	3.7,17	
sqlite-libs		3,40,0
system-release	2	2023,4.20240513
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	
zlib	1.2.7	1.2.11

Comparaison entre AL1 et AL2023

Les rubriques suivantes décrivent les principales différences entre AL1 et AL2023 qui ne sont pas déjà couvertes par la comparaison avec AL2.

> **Note**
>
> AL1 a atteint son end-of-life (EOL) le 31 décembre 2023 et ne recevra aucune mise à jour de sécurité ni aucune correction de bogue à compter du 1er janvier 2024. Pour plus d'informations sur AL1 EOL et le support de maintenance, consultez le billet de blog Update on Amazon Linux AMI. end-of-life Nous vous recommandons de mettre à niveau les applications vers AL2023, qui inclut un support à long terme jusqu'en 2028.

Rubriques

Support pour chaque version

Pour AL2023, nous proposons cinq ans de support à compter de la date de sortie. AL1 a mis fin au support standard le 31 décembre 2020 et a mis fin au support de maintenance au 31 décembre 2023.

Pour plus d'informations, consultez Cadence de publication.

`systemd` remplace `upstart` en tant que système `init`

Dans AL2, `upstart` il a été remplacé par `systemd` as the `init` system. AL2023 utilise également `systemd` comme `init` système, adoptant en outre de nouvelles fonctionnalités de`systemd`.

Python 2.6 et 2.7 ont été remplacés par Python 3

Bien qu'AL1 ait marqué Python 2.6 comme EOL dans la version 2018.03, les packages étaient toujours disponibles dans les référentiels à installer. AL2 a été livré avec Python 2.7 comme première version de Python prise en charge, et AL2023 complète la transition vers Python 3. Aucune version de Python 2.x n'est incluse dans les référentiels AL2023.

Pour plus d'informations relatives à Python sur Amazon Linux, consultez Python dans AL2023.

OpenJDK 8 en tant que plus ancien JDK

AL2023 est livré avec Amazon Corretto comme kit de développement Java (JDK) par défaut (et unique). Tous les packages Java basés dans AL2023 sont construits avecAmazon Corretto 17.

Dans AL1, OpenJDK 1.6.0 `java-1.6.0-openjdk` () est passé en EOL avec la première version 2018.03, et OpenJDK 1.7.0 `java-1.7.0-openjdk` () est passé en EOL à la mi-2020, bien que les deux versions soient disponibles dans les référentiels AL1. La première version d'OpenJDK disponible en AL2023 est OpenJDK 8, fournie par. Amazon Corretto 8

Modifications apportées au noyau AL2023 par rapport à Amazon Linux 1 (AL1)

Kernel Live Patching

AL2023 et AL2 ajoutent tous deux la prise en charge de la fonctionnalité de mise à jour dynamique du noyau. Cela vous permet de corriger les failles de sécurité critiques et importantes du noyau Linux sans redémarrage ni interruption de service. Pour plus d'informations, consultez Kernel Live Patching sur AL2023.

Prise en charge des systèmes de fichiers par le noyau

Plusieurs modifications ont été apportées aux systèmes de fichiers que le noyau d'AL1 permettra de monter, ainsi que des modifications dans les schémas de partitionnement que le noyau analysera.

Option **CONFIG**	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_AFS_FS	m	n	n
CONFIG_AF_RXRPC	m	n	n
CONFIG_BSD_DISKLABEL	y	n	n
CONFIG_CRAMFS	m	n	n
CONFIG_CRAMFS_BLOCKDEV	N/A	N/A	N/A
CONFIG_DM_CLONE	N/A	n	n
CONFIG_DM_ERA	n	n	n
CONFIG_DM_INTEGRITY	m	m	m
CONFIG_DM_LOG_WRITES	n	m	m
CONFIG_DM_SWITCH	n	n	n
CONFIG_DM_VERITY	n	n	n
CONFIG_ECRYPT_FS	m	n	n
CONFIG_EXFAT_FS	N/A	m	m
CONFIG_EXT2_FS	m	n	n
CONFIG_EXT3_FS	m	n	n
CONFIG_GFS2_FS	n	n	n

Option **CONFIG**	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_HFSPLUS_FS	m	n	n
CONFIG_HFS_FS	m	n	n
CONFIG_JFS_FS	m	n	n
CONFIG_LDM_PARTITION	y	n	n
CONFIG_MAC_PARTITION	y	n	n
CONFIG_NFS_V2	m	n	n
CONFIG_NTFS_FS	m	n	n
CONFIG_ROMFS_FS	m	n	n
CONFIG_SOLARIS_X86_PARTITION	y	n	n
CONFIG_SQUASHFS_ZSTD	y	y	y
CONFIG_SUN_PARTITION	y	n	n

Modifications de configuration du noyau axées sur la sécurité

Option **CONFIG**	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_BUG_ON_DATA_CORRUPTION	y	y	y

Option **CONFIG**	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_DEFAULT_MMAP_MIN_ADDR	4096	65536	65536
CONFIG_DEVMEM	y	n	n
CONFIG_DEVPORT	y	n	n
CONFIG_FORTIFY_SOURCE	y	y	y
CONFIG_HARDENED_USERCOPY_FALLBACK	N/A	N/A	N/A
CONFIG_INIT_ON_ALLOC_DEFAULT_ON	N/A	n	n
CONFIG_INIT_ON_FREE_DEFAULT_ON	N/A	n	n
CONFIG_IOMMU_DEFAULT_DMA_STRICT	N/A	n	n
CONFIG_LDISC_AUTOLOAD	y	n	n
CONFIG_SCHED_CORE	N/A	N/A	y
CONFIG_SCHED_STACK_END_CHECK	y	y	y

Option **CONFIG**	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_SECURITY_DMESG_RESTRICT	n	y	y
CONFIG_SECURITY_SELINUX_DISABLE	y	n	n
CONFIG_SHUFFLE_PAGE_ALLOCATOR	N/A	y	y
CONFIG_SLAB_FREELIST_HARDENED	y	y	y
CONFIG_SLAB_FREELIST_RANDOM	n	y	y

Autres modifications de configuration du noyau

Option **CONFIG**	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_HZ	250	100	100
CONFIG_NR_CPUS	8192	512	512
CONFIG_PANIC_ON_OOPS	n	y	y
CONFIG_PANIC_ON_OOPS_VALUE	0	1	1

Option **CONFIG**	AL1/4.14/x86_64	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_PPP	m	n	n
CONFIG_SLIP	m	n	n
CONFIG_XEN_PV	y	N/A	n

Comparaison des packages installés sur les AMI Amazon Linux 1 (AL1) et Amazon Linux 2023

Comparaison des RPM présents sur les AMI standard AL1 et AL2023.

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
acl	2,2,49	2.3.1
acpid	2,0,19	2,0,32
alsa-lib	1,0,22	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.4.1
amazon-linux-repo-s3		2023,4.20240513
amazon-linux-sb-keys		2023.1
amazon-rpm-config		228
amazon-ssm-agent	3.2.2222.0	3,3.380,0
at	3.1.10	3,1,23
attr	2,4,46	2.5.1

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
bc	1,06,95	1,07.1
bind-libs	9.8.2	9,16,48
bind-license		9,16,48
bind-utils	9.8.2	9,16,48
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
ca-certificates	2023,2,62	2023,2,64
c-ares		1.19.0
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	
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		20220428
crypto-policies-scripts		20220428

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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	2,127
cyrus-sasl-plain	2.1.23	2,127
dash	0,5.5.1	
db4	4,7,25	
db4-utils	4,7,25	
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	

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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	
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
hunspell-filesystem		1.7.0
hwdata	0,233	0,353
info	5.1	6.7
inih		49
initscripts	9,03,58	10,09
iproute	4.4.0	5.10.0
iptables	1.4,21	
iputils	20121221	20210202
irqbalance	1.5.0	1.9.0
jansson		2.14
java-1.7.0-openjdk	1,7,0,321	
javapackages-tools	0.9.1	
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	6,1,90
kernel-livepatch-repo-s3		2023,4.20240513

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
kernel-srpm-macros		1.0
kernel-tools	4,14.336	6,1,90
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
krb5-libs	1.15.1	1,21
lcms2	2.6	
less	436	608
libacl	2,2,49	2.3.1
libaio	0,3,109	0,3,111
libarchive		3.5.3
libargon2		27/12/2017
libassuan	2.0.3	2.5.5
libattr	2,4,46	2.5.1
libbasicobjects		0,11
libblkid	2.23.2	2,37,4
libcap	2,16	2,48

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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		0,13
libmnl	1.0.3	1.0.4

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
libmodulemd		2.13.0
libmount	2.23.2	2,37,4
libnetfilter_connt rack	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	
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		0,15
librepo		1.14,5
libreport-filesystem		2.15.2
libseccomp		2.5.3

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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		4.4.33
libXext	1.3.2	
libXfont	1.4.5	

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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	0,16	0,2,5
libzstd		1.5.5
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	

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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	
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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	7,4 p1	8,7 p1
openssh-clients	7,4 p1	8,7 p1
openssh-server	7,4 p1	8,7 p1
openssl	1,02k	3,0.8
openssl-libs		3,0.8

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
openssl-pkcs11		0,4,12
os-prober		1,77
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
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		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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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		2021,0726
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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	1,1,31	
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	

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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	
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	0,17	
python27-pycurl	7,19,0	

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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	
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	15,10	
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
python3-libs		3,9,16
python3-libselinux		3.4
python3-libsemanage		3.4
python3-libstorage mgmt		1.9.4
python3-lockfile		0.12.2
python3-markupsafe		1.1.1
python3-netifaces		0,1,6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1
python3-ply		3,11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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		0,12
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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		4.16.1.3
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	

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
rust-srpm-macros		21
sbsigntools		0.9.4
screen	4.0.3	4.8.0
sed	4.2.1	4.8
selinux-policy		37,22
selinux-policy-targeted		37,22
sendmail	8,1,4	
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
sudo	1,8,23	1,9,15
sysctl-defaults	1.0	1.0
sysfsutils	2.1.0	
sysstat		12,5.6
systemd		252,16
systemd-libs		252,16
systemd-networkd		252,16
systemd-pam		252,16
systemd-resolved		252,16
systemd-udev		252,16
system-release	2018,03	2023,4.20240513
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		6,24,07
time	1,7	1.9

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
tmpwatch	2.9,16	
traceroute	2,0,14	2.1.3
ttmkfdir	3.0.9	
tzdata	2023c	2024a
tzdata-java	2023c	
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
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	
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

Package	TOUTES LES 1 AMI	TOUS LES 2023 AMI
zstd		1.5.5

Comparaison des packages installés sur les AMI minimales Amazon Linux 1 (AL1) et Amazon Linux 2023

Comparaison des RPM présents sur les AMI minimales AL1 et AL2023.

Package	AL1 Minimale	AL2023 Minimale
acpid	2,0,19	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.4.1
amazon-linux-repo-s3		2023,4.20240513
amazon-linux-sb-keys		2023.1
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	

Package	AL1 Minimale	AL2023 Minimale
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023,2,62	2023,2,64
checkpolicy	2.1.10	3.4
chkconfig	1.3.49,3	
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
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		20220428
cryptsetup-libs		2.6.1
curl	7,61,1	

Package	AL1 Minimale	AL2023 Minimale
curl-minimal		8.5.0
cyrus-sasl	2.1.23	
cyrus-sasl-lib	2.1.23	2,127
dash	0,5.5.1	
db4	4,7,25	
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

Package	AL1 Minimale	AL2023 Minimale
dnf-plugin-support-info		1.2
dracut	004	055
dracut-config-ec2		3.0
dracut-config-generic		055
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

Package	AL1 Minimale	AL2023 Minimale
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	
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

Package	AL1 Minimale	AL2023 Minimale
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
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	0,353
info	5.1	

Package	AL1 Minimale	AL2023 Minimale
inih		49
initscripts	9,03,58	10,09
iproute	4.4.0	5.10.0
iptables	1.4,21	
iputils	20121221	20210202
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	6,1,90
kernel-livepatch-repo-s3		2023,4.20240513
keyutils-libs	1.5.8	1.6.3
kmod	14	29
kmod-libs	14	29
krb5-libs	1.15.1	1,21
less	436	608
libacl	2,2,49	2.3.1

Package	AL1 Minimale	AL2023 Minimale
libarchive		3.5.3
libargon2		27/12/2017
libassuan	2.0.3	2.5.5
libattr	2,4,46	2.5.1
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	
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

Package	AL1 Minimale	AL2023 Minimale
libfido2		1.10.0
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	
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

Package	AL1 Minimale	AL2023 Minimale
libpwquality	1.2.3	1.4.4
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
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

Package	AL1 Minimale	AL2023 Minimale
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
libxcrypt		4.4.33
libxml2	2.9.1	2.10.4
libyaml	0,16	0,2,5
libzstd		1.5.5
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

Package	AL1 Minimale	AL2023 Minimale
nettle		3.8
net-tools	1,60	2.0
newt	0,52,11	
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	7,4 p1	8,7 p1
openssh-clients		8,7 p1

Package	AL1 Minimale	AL2023 Minimale
openssh-server	7,4 p1	8,7 p1
openssl	1,02k	3,0.8
openssl-libs		3,0.8
openssl-pkcs11		0,4,12
os-prober		1,77
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		10,40
pcre2-syntax		10,40
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	

Package	AL1 Minimale	AL2023 Minimale
procps-ng		3.3,17
psmisc	22,20	23,4
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-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	

Package	AL1 Minimale	AL2023 Minimale
python27-pyxattr	0,5,0	
python27-PyYAML	3,10	
python27-requests	1.2.3	
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

Package	AL1 Minimale	AL2023 Minimale
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-markupsafe		1.1.1
python3-netifaces		0,1,6
python3-oauthlib		3.0.2
python3-pip-wheel		21.3.1

Package	AL1 Minimale	AL2023 Minimale
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		0,12
python3-setools		4.4.1
python3-setuptools		59,6,0
python3-setuptools-wheel		59,6,0
python3-six		1.15.0

Package	AL1 Minimale	AL2023 Minimale
python3-systemd		235
python3-urllib3		1,25,10
python3-wcwidth		0,2,5
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		37,22
selinux-policy-targeted		37,22
sendmail	8,1,4	

Package	AL1 Minimale	AL2023 Minimale
setserial	2,17	
setup	2.8,14	2.13.7
shadow-utils	4.1.4.2	4,9
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		252,16
systemd-libs		252,16
systemd-networkd		252,16
systemd-pam		252,16
systemd-resolved		252,16
systemd-udev		252,16
system-release	2018,03	2023,4.20240513
sysvinit	2,87	
tar	1,26	1,34
tcp_wrappers-libs	7.6	

Package	AL1 Minimale	AL2023 Minimale
tzdata	2023c	2024a
udev	173	
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-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

Package	AL1 Minimale	AL2023 Minimale
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1.5.5

Comparaison des packages installés sur les images de conteneurs de base Amazon Linux 1 (AL1) et Amazon Linux 2023

Comparaison des RPM présents sur les images des conteneurs de base AL1 et AL2023.

Package	Conteneur AL1	Conteneur AL2023
alternatives		1.15
amazon-linux-repo-cdn		2023,4.20240513
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	2023,2,64
chkconfig	1.3.49,3	
coreutils	8,22	
coreutils-single		8,32
crypto-policies		20220428

Package	Conteneur AL1	Conteneur AL2023
curl	7,61,1	
curl-minimal		8.5.0
cyrus-sasl-lib	2.1.23	
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

Package	Conteneur AL1	Conteneur AL2023
glibc-minimal-langpack		2,34
gmp	6.0.0	6.2.1
gnupg2	2,0,28	
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
libacl	2,2,49	2.3.1
libarchive		3.5.3
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

Package	Conteneur AL1	Conteneur AL2023
libcurl	7,61,1	
libcurl-minimal		8.5.0
libdnf		0,69,0
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

Package	Conteneur AL1	Conteneur AL2023
libsmartcols		2,37,4
libsolv		0,7,22
libssh2	1.4.2	
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		4.4.33
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	

Package	Conteneur AL1	Conteneur AL2023
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
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	
openldap	2,4,40	
openssl	1,02k	
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		10,40
pcre2-syntax		10,40
pinentry	0,7,6	

Package	Conteneur AL1	Conteneur AL2023
pkgconfig	0,27,1	
popt	1.13	1,18
pth	2.0.7	
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

Package	Conteneur AL1	Conteneur AL2023
python3-libs		3,9,16
python3-pip-wheel		21.3.1
python3-rpm		4.16.1.3
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	2023,4.20240513
tar	1,26	
tzdata	2023c	2024a
xz-libs	5.2.2	5.2.5

Package	Conteneur AL1	Conteneur AL2023
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	
yum-utils	1.1.31	
zlib	1.2.8	1.2.11

Configuration système requise pour AL2023

Cette section décrit la configuration système requise pour utiliser AL2023.

Rubriques

Configuration du processeur requise pour exécuter AL2023

Pour exécuter n'importe quel code AL2023, le processeur utilisé doit répondre à certaines exigences minimales. Les tentatives d'exécution d'AL2023 sur des processeurs qui ne répondent pas à ces exigences peuvent entraîner des erreurs d'instructions illégales très tôt dans l'exécution du code.

Les exigences minimales s'appliquent à AL2023 sur Amazon EC2AL2023 dans des conteneurs, etAL2023 en dehors d'Amazon EC2.

Exigences liées au processeur ARM pour AL2023

Tous les fichiers binaires AL2023 `aarch64` (ARM) sont conçus pour 64 bits. Aucun fichier ARM binaire 32 bits n'étant disponible, un ARM processeur 64 bits est requis.

> **Note**
>
> Pour les instances basées sur ARM, AL2023 prend en charge uniquement les types d'instance qui utilisent des processeurs Graviton2 ou ultérieurs. AL2023 ne prend pas en charge les instances A1.

L'AL2023 nécessite un processeur compatible ARMv8.2 avec l'extension cryptographique (`ARMv8.2+crypto`). Tous les packages AL2023 pour `aarch64` sont créés avec l'indicateur du `-march=armv8.2-a+crypto` compilateur. Bien que nous essayions d'imprimer des messages d'erreur élégants lorsque le code AL2023 est tenté d'être exécuté sur des ARM processeurs plus anciens, il est possible que le premier message d'erreur soit une erreur d'instruction illégale.

> **Note**
>
> En raison des exigences de `aarch64` base du processeur AL2023, tous les Raspberry Pi systèmes antérieurs Raspberry Pi 5 ne répondent pas aux exigences minimales en matière de processeur.

Exigences liées au processeur x86-64 pour AL2023

Tous les `x86-64` binaires AL2023 sont conçus pour la `x86-64v2` révision de l'`x86-64`architecture en passant `-march=x86-64-v2` au compilateur.

La `x86-64v2` révision de l'architecture ajoute les fonctionnalités de processeur suivantes en plus de l'`x86-64`architecture de base :

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

Cela correspond approximativement aux `x86-64` processeurs sortis en 2009 ou ultérieurement. Les exemples incluent les microarchitectures Intel NehalemAMD Jaguar,Atom Silvermont,, ainsi que les Eden C microarchitectures VIA Nano et.

Dans Amazon EC2, tous les types d'instances `x86-64` prennent en charge `x86-64v2`, notamment `M1`, `C1` et les familles d'instances `M2`.

Aucun fichier binaire 32 bits x86 (i686) AL2023 n'est créé. Bien qu'AL2023 continue de prendre en charge l'exécution de fichiers binaires 32 bits en espace utilisateur, cette fonctionnalité est obsolète et pourrait être supprimée dans une future version majeure d'Amazon Linux. Pour plus d'informations, consultez Packages x86 (i686) 32 bits.

Exigences en matière de mémoire (RAM) pour exécuter AL2023

La `.nano` famille de types d'instances Amazon EC2 (`t2.nano`,`t3.nano`, et`t4g.nano`) dispose de 512 Mo de RAM`t3a.nano`, ce qui est la configuration minimale requise pour AL2023.

> **Note**
>
> Bien que 512 Mo soient le minimum requis, ces types d'instances sont soumis à des contraintes de mémoire et leurs fonctionnalités et performances peuvent être limitées.

Les images AL2023 n'ont pas été testées sur des systèmes dotés de moins de 512 Mo de RAM. L'exécution d'images de conteneur basées sur AL2023 dans moins de 512 Mo de RAM dépendra de la charge de travail conteneurisée.

Certaines charges de travail, par exemple `dnf update` entre certaines versions d'AL2023, peuvent nécessiter plus de 512 Mo de RAM. C'est pourquoi la version AL2023.3 a introduit l'activation `zram` par défaut pour les instances disposant de moins de 800 Mo de RAM. Pour les charges de travail conteneurisées, cela signifie que certaines charges de travail peuvent fonctionner correctement sur des instances AL2023 avec cette quantité de mémoire, mais échouer lorsqu'elles sont exécutées dans un conteneur limité à cette quantité de mémoire.

Pour les types d'instance disposant de moins de 800 Mo de RAM, AL2023 (à partir d'AL2023.3 ou plus récent) active l'échange basé sur `zram` par défaut. Les exemples de types d'instances Amazon EC2 avec moins de 800 Mo de mémoire incluent`t4g.nano`,, `t3a.nano` `t3.nano`t2.nano, et. `t1.micro` Cela signifie moins de scénarios de mémoire insuffisante pour ces types d'instance, car AL2023 compresse et décompresse les pages de mémoire à la demande. Cela permet d'activer des charges de travail qui nécessiteraient autrement un type d'instance doté de plus de mémoire, au détriment de l'utilisation du processeur nécessaire à la compression.

Utilisation d'AL2023 sur AWS

Vous pouvez configurer AL2023 pour l'utiliser avec d'autres Services AWS. Par exemple, vous pouvez choisir une AMI AL2023 lorsque vous lancez une instance Amazon Elastic Compute Cloud (Amazon EC2).

Pour ces procédures de configuration, vous utilisez le service AWS Identity and Access Management (IAM). Pour obtenir des informations complètes sur IAM, consultez les documents de référence suivants :

- AWS Identity and Access Management (JE SUIS)
- Guide de l'utilisateur IAM

Rubriques

Commencer avec AWS

Inscrivez-vous pour un Compte AWS

Si vous n'en avez pas Compte AWS, procédez comme suit pour en créer un.

Pour vous inscrire à un Compte AWS

1. Ouvrez https://portal.aws.amazon.com/billing/signup.
2. Suivez les instructions en ligne.

Dans le cadre de la procédure d'inscription, vous recevrez un appel téléphonique et vous saisirez un code de vérification en utilisant le clavier numérique du téléphone.

Lorsque vous vous inscrivez à un Compte AWS, un Utilisateur racine d'un compte AWSest créé. Par défaut, seul l'utilisateur racine a accès à l'ensemble des Services AWS et des ressources de ce compte. La meilleure pratique en matière de sécurité consiste à attribuer un accès administratif à un utilisateur et à n'utiliser que l'utilisateur root pour effectuer les tâches nécessitant un accès utilisateur root.

AWS vous envoie un e-mail de confirmation une fois le processus d'inscription terminé. Vous pouvez afficher l'activité en cours de votre compte et gérer votre compte à tout moment en accédant à https://aws.amazon.com/ et en choisissant Mon compte.

Création d'un utilisateur doté d'un accès administratif

Une fois que vous vous êtes inscrit à un utilisateur administratif Compte AWS, que vous Utilisateur racine d'un compte AWS l'avez sécurisé AWS IAM Identity Center, que vous l'avez activé et que vous en avez créé un, afin de ne pas utiliser l'utilisateur root pour les tâches quotidiennes.

Sécurisez votre Utilisateur racine d'un compte AWS

1. Connectez-vous en AWS Management Consoletant que propriétaire du compte en choisissant Utilisateur root et en saisissant votre adresse Compte AWS e-mail. Sur la page suivante, saisissez votre mot de passe.

   Pour obtenir de l'aide pour vous connecter en utilisant l'utilisateur racine, consultez Connexion en tant qu'utilisateur racine dans le Guide de l'utilisateur Connexion à AWS .
2. Activez l'authentification multifactorielle (MFA) pour votre utilisateur racine.

   Pour obtenir des instructions, voir Activer un périphérique MFA virtuel pour votre utilisateur Compte AWS root (console) dans le guide de l'utilisateur IAM.

Création d'un utilisateur doté d'un accès administratif

1. Activez IAM Identity Center.

   Pour obtenir des instructions, consultez Activation d' AWS IAM Identity Center dans le Guide de l'utilisateur AWS IAM Identity Center .

2. Dans IAM Identity Center, accordez un accès administratif à un utilisateur.

   Pour un didacticiel sur l'utilisation du Répertoire IAM Identity Center comme source d'identité, voir Configurer l'accès utilisateur par défaut Répertoire IAM Identity Center dans le Guide de AWS IAM Identity Center l'utilisateur.

Connectez-vous en tant qu'utilisateur disposant d'un accès administratif

- Pour vous connecter avec votre utilisateur IAM Identity Center, utilisez l'URL de connexion qui a été envoyée à votre adresse e-mail lorsque vous avez créé l'utilisateur IAM Identity Center.

  Pour obtenir de l'aide pour vous connecter en utilisant un utilisateur d'IAM Identity Center, consultez la section Connexion au portail AWS d'accès dans le guide de l'Connexion à AWS utilisateur.

Attribuer l'accès à des utilisateurs supplémentaires

1. Dans IAM Identity Center, créez un ensemble d'autorisations conforme aux meilleures pratiques en matière d'application des autorisations du moindre privilège.

   Pour obtenir des instructions, voir Création d'un ensemble d'autorisations dans le guide de AWS IAM Identity Center l'utilisateur.
2. Affectez des utilisateurs à un groupe, puis attribuez un accès d'authentification unique au groupe.

   Pour obtenir des instructions, voir Ajouter des groupes dans le guide de AWS IAM Identity Center l'utilisateur.

Accorder un accès par programmation

Les utilisateurs ont besoin d'un accès programmatique s'ils souhaitent interagir avec AWS l'extérieur du AWS Management Console. La manière d'accorder un accès programmatique dépend du type d'utilisateur qui y accède AWS.

Pour accorder aux utilisateurs un accès programmatique, choisissez l'une des options suivantes.

Quel utilisateur a besoin d'un accès programmatique ?	Pour	Par
Identité de la main-d'œuvre (Utilisateurs gérés dans IAM Identity Center)	Utilisez des informations d'identification temporaires pour signer les demandes programmatiques adressées aux AWS CLI AWS SDK ou AWS aux API.	Suivez les instructions de l'interface que vous souhaitez utiliser. • Pour le AWS CLI, voir Configuration du AWS CLI à utiliser AWS IAM Identity Center dans le guide de AWS Command Line Interface l'utilisateur. • Pour les AWS SDK, les outils et les AWS API, consultez la section Authentification IAM Identity Center dans le Guide de référence AWS des SDK et des outils.
IAM	Utilisez des informations d'identification temporaires pour signer les demandes programmatiques adressées aux AWS CLI AWS SDK ou AWS aux API.	Suivez les instructions de la section Utilisation d'informations d'identification temporaires avec AWS les ressources du Guide de l'utilisateur IAM.
IAM	(Non recommandé) Utilisez des informations d'identification à long terme pour signer les AWS CLI demandes programmatiques adressées aux AWS SDK ou AWS aux API.	Suivez les instructions de l'interface que vous souhaitez utiliser. • Pour le AWS CLI, voir Authentification à l'aide des informations d'identification utilisateur IAM dans le

Quel utilisateur a besoin d'un accès programmatique ?	Pour	Par
		Guide de l'AWS Command Line Interface utilisateur. • Pour les AWS SDK et les outils, voir Authentifier à l'aide d'informations d'identification à long terme dans le Guide de AWS référence des SDK et des outils. • Pour les AWS API, consultez la section Gestion des clés d'accès pour les utilisateurs IAM dans le guide de l'utilisateur IAM.

AL2023 sur Amazon EC2

Utilisez l'une des procédures suivantes pour lancer une instance Amazon EC2 avec une AMI AL2023. Vous pouvez choisir l'AMI standard ou l'AMI minimale. Pour plus d'informations sur les différences entre l'AMI standard et l'AMI minimale, consultez Comparaison entre les AMI standard (par défaut) et minimale d'AL2023.

Rubriques

- Lancement d'AL2023 à l'aide de la console Amazon EC2
- Lancement d'AL2023 à l'aide du paramètre SSM et AWS CLI
- Lancement de la dernière AMI AL2023 en utilisant AWS CloudFormation
- Lancement d'AL2023 à l'aide d'un ID AMI spécifique
- Dépréciation et cycle de vie de l'AMI AL2023
- Connexion aux instances AL2023
- Comparaison entre la norme AL2023 et les AMI minimales

Lancement d'AL2023 à l'aide de la console Amazon EC2

Utilisez la console Amazon EC2 pour lancer une AMI AL2023.

> **Note**
>
> Pour les instances basées sur ARM, AL2023 prend en charge uniquement les types d'instance qui utilisent des processeurs Graviton2 ou ultérieurs. AL2023 ne prend pas en charge les instances A1.

Utilisez les étapes suivantes pour lancer une instance Amazon EC2 avec une AMI AL2023 à partir de la console Amazon EC2.

Pour lancer une instance EC2 avec une AMI AL2023

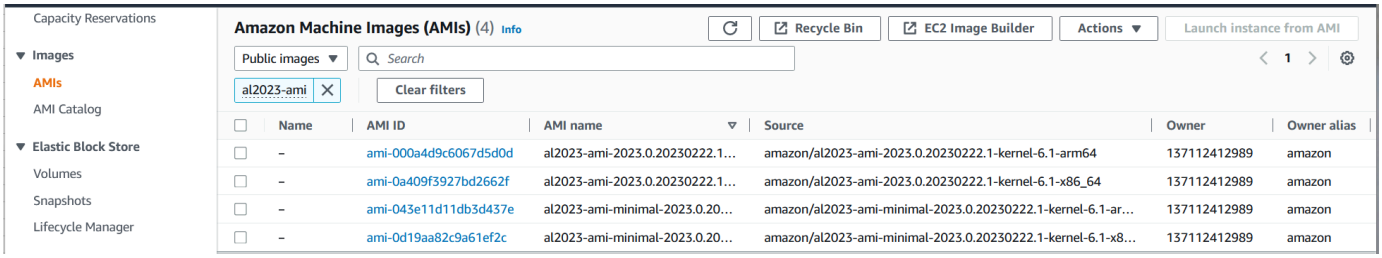
1. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, sélectionnez AMI.
3. Dans la liste déroulante de filtre, choisissez Images publiques.
4. Dans le champ de recherche, entrez **`al2023-ami`**.

   > **Note**
   >
   > Assurez-vous qu'amazon apparaît dans la colonne Alias du propriétaire.

5. Sélectionnez une image dans la liste. Sous Source, vous pouvez déterminer si l'AMI est standard ou minimale. Le nom d'une AMI AL2023 peut être interprété en utilisant ce format :

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

6. L'image suivante montre une liste partielle des AMI AL2023.



Pour plus d'informations sur le lancement d'instances Amazon EC2, consultez Commencer avec les instances Amazon EC2 Linux dans le guide de l'utilisateur Amazon EC2.

Lancement d'AL2023 à l'aide du paramètre SSM et AWS CLI

Dans le AWS CLI, vous pouvez utiliser la valeur du paramètre SSM d'une AMI pour lancer une nouvelle instance d'AL2023. Plus précisément, utilisez l'une des valeurs de paramètre SSM dynamiques de la liste suivante, et ajoutez `/aws/service/ami-amazon-linux-latest/` avant la valeur du paramètre SSM. Vous utilisez cela pour lancer l'instance dans AWS CLI.

- `al2023-ami-kernel-default-arm64` pour l'architecture arm64
- `al2023-ami-minimal-kernel-default-arm64` pour l'architecture arm64 (AMI minimale)
- `al2023-ami-kernel-default-x86_64` pour l'architecture x86_64
- `al2023-ami-minimal-kernel-default-x86_64` pour l'architecture x86_64 (AMI minimale)

> **Note**
>
> Chacun des éléments *`en italique`* est un exemple de paramètre. Remplacez-les par vos propres informations.

```
$ 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
```

L'indicateur `--image-id` spécifie la valeur du paramètre SSM.

L'indicateur `--instance-type` spécifie le type et la taille de l'instance. Cet indicateur doit être compatible avec le type d'AMI que vous avez sélectionné.

Le `--region` drapeau indique l' Région AWS endroit où vous créez votre instance.

L'`--key-name`indicateur indique Région AWS la clé utilisée pour se connecter à l'instance. Si vous ne fournissez pas de clé qui existe dans la région où vous créez l'instance, vous ne pouvez pas vous connecter à l'instance via SSH.

L'indicateur `--security-group-ids` spécifie le groupe de sécurité qui détermine les autorisations d'accès pour le trafic réseau entrant et sortant.

> **Important**
>
> Vous devez spécifier un groupe de sécurité existant qui autorise l'accès à l'instance depuis votre machine distante via le portTCP:22. AWS CLI Sans groupe de sécurité spécifié, votre nouvelle instance est placée dans un groupe de sécurité par défaut. Dans un groupe de sécurité par défaut, votre instance ne peut se connecter qu'aux autres instances de votre VPC.

Pour plus d'informations, consultez Lancement, mise en liste et arrêt des instances Amazon EC2 dans le Guide de l'utilisateur AWS Command Line Interface .

Lancement de la dernière AMI AL2023 en utilisant AWS CloudFormation

Pour lancer une AMI AL2023 à l'aide de AWS CloudFormation, utilisez l'un des modèles suivants.

> **Note**
>
> Les AMI `x86_64` et `Arm64` nécessitent chacune des types d'instance différents. Pour plus d'informations, consultez la rubrique Types d'instance Amazon EC2.

Modèle `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"
        }
      }
    }
  }
}
```

Modèle 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
```

Veillez à remplacer le paramètre AMI à la fin de la section « Par défaut », si nécessaire. Les valeurs suivantes de paramètre sont disponibles :

- `al2023-ami-kernel-6.1-arm64` pour l'architecture arm64
- `al2023-ami-minimal-kernel-6.1-arm64` pour l'architecture arm64 (AMI minimale)
- `al2023-ami-kernel-6.1-x86_64` pour l'architecture x86_64
- `al2023-ami-minimal-kernel-6.1-x86_64` pour l'architecture x86_64 (AMI minimale)

Les spécifications du noyau dynamique sont indiquées ci-dessous. La version par défaut du noyau change automatiquement à chaque mise à jour majeure de la version du noyau.

- `al2023-ami-kernel-default-arm64` pour l'architecture arm64
- `al2023-ami-minimal-kernel-default-arm64` pour l'architecture arm64 (AMI minimale)
- `al2023-ami-kernel-default-x86_64` pour l'architecture x86_64
- `al2023-ami-minimal-kernel-default-x86_64` pour l'architecture x86_64 (AMI minimale)

Lancement d'AL2023 à l'aide d'un ID AMI spécifique

Vous pouvez lancer une AMI AL2023 spécifique à l'aide de l'ID de l'AMI. Vous pouvez déterminer quel ID d'AMI AL2023 est nécessaire en consultant la liste des AMI dans la console Amazon EC2. Ou, vous pouvez utiliser AWS Systems Manager. Si vous utilisez Systems Manager, veillez à sélectionner l'alias d'AMI parmi ceux répertoriés dans la section précédente. Pour plus d'informations, consultez la section Rechercher les derniers ID d'AMI Amazon Linux à l'aide de AWS Systems Manager Parameter Store.

Dépréciation et cycle de vie de l'AMI AL2023

Chaque nouvelle version d'AL2023 inclut une nouvelle AMI. Quand l'AMI est enregistrée, elle est marquée d'une date d'obsolescence. La date d'obsolescence de chaque AMI AL2023 est le 90e jour après sa publication, pour être conforme à la durée de l'offre Kernel Live Patching sur AL2023 pour chaque version individuelle du noyau.

> **Note**
>
> Le délai d'obsolescence de 90 jours fait référence à une AMI individuelle et ne se rapporte pas à la Cadence de publication d'AL2023 ni à la période de support du produit.

Pour plus d'informations sur la dépréciation d'une AMI, consultez la section Dépréciation d'une AMI dans le guide de l'utilisateur Amazon EC2.

L'utilisation régulière d'une AMI mise à jour pour lancer une instance garantit que celle-ci démarre avec les dernières mises à jour de sécurité, notamment un noyau mis à jour. Si vous lancez une version précédente d'une AMI et appliquez des mises à jour, l'instance ne dispose pas des dernières mises à jour de sécurité pendant un certain temps. Pour vous assurer d'utiliser la dernière AMI, nous vous recommandons d'utiliser les paramètres SSM.

Pour plus d'informations sur l'utilisation des paramètres SSM pour lancer une instance, consultez :

- Lancement d'AL2023 à l'aide du paramètre SSM et AWS CLI
- Lancement de la dernière AMI AL2023 en utilisant AWS CloudFormation

Connexion aux instances AL2023

Utilisez SSH ou AWS Systems Manager pour vous connecter à votre instance AL2023.

Se connecter à votre instance à l'aide de SSH

Pour obtenir des instructions sur la façon d'utiliser SSH pour se connecter à une instance, consultez Connect to your Linux instance using SSH dans le guide de l'utilisateur Amazon EC2.

Connectez-vous à votre instance à l'aide de AWS Systems Manager

Pour obtenir des instructions sur la manière de se connecter AWS Systems Manager à une instance AL2023, consultez Se connecter à votre instance Linux à l'aide du gestionnaire de session dans le guide de l'utilisateur Amazon EC2.

Utilisation d'Amazon EC2 Instance Connect

L'AMI AL2023, à l'exception de l'AMI minimale, est fournie avec l'agent EC2 Instance Connect installé par défaut. Pour utiliser EC2 Instance Connect avec une instance AL2023 lancée à partir de l'AMI minimale, vous devez installer le `ec2-instance-connect` package. Pour obtenir des instructions sur l'utilisation d'EC2 Instance Connect, consultez la section Se connecter à votre instance Linux avec EC2 Instance Connect dans le guide de l'utilisateur Amazon EC2.

Comparaison entre la norme AL2023 et les AMI minimales

Vous pouvez lancer une instance Amazon EC2 avec une AMI AL2023 standard (par défaut) ou minimale. Pour savoir comment lancer une instance Amazon EC2 avec le type d'AMI standard ou minimal, consultez. AL2023 sur Amazon EC2

L'AMI AL2023 standard est livré avec toutes les applications et tous les outils les plus couramment utilisés installés. Nous recommandons l'AMI standard si vous souhaitez démarrer rapidement et que vous n'êtes pas intéressé par la personnalisation de l'AMI.

L'AMI AL2023 minimale est la version de base simplifiée qui ne contient que les outils et utilitaires les plus élémentaires nécessaires pour exécuter le système d'exploitation (OS). Nous vous recommandons d'utiliser l'AMI minimale si vous souhaitez réduire au maximum l'encombrement du système d'exploitation. L'AMI minimale permet de réduire légèrement l'utilisation de l'espace disque et d'améliorer la rentabilité à long terme. L'AMI minimale convient si vous souhaitez un système d'exploitation plus petit et que cela ne vous dérange pas d'installer manuellement des outils et des applications.

L'image du conteneur est plus proche de l'AMI minimale AL2023 dans l'ensemble des packages.

Comparaison des packages installés sur les images Amazon Linux 2023

Comparaison des RPM présents sur les images de l'AMI AL2023, de l'AMI minimale et du conteneur.

Package	AMI	AMI minimale	Conteneur
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	2.4.1	2.4.1	
amazon-linux-repo-cdn			2023,4.20240513
amazon-linux-repo-s3	2023,4.20240513	2023,4.20240513	
amazon-linux-sb-keys	2023.1	2023.1	
amazon-rpm-config	228		
amazon-ssm-agent	3,3.380,0		
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		

Package	AMI	AMI minimale	Conteneur
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	9,16,48		
bind-license	9,16,48		
bind-utils	9,16,48		
binutils	2,39		
boost-filesystem	1,75,0		
boost-system	1,75,0		
boost-thread	1,75,0		
bzip2	1.0.8		
bzip2-libs	1.0.8	1.0.8	1.0.8
ca-certificates	2023,2,64	2023,2,64	2023,2,64
c-ares	1.19.0		
checkpolicy	3.4	3.4	
chkconfig	1.15		
chrony	4.3	4.3	
cloud-init	22.2.2	22.2.2	

Package	AMI	AMI minimale	Conteneur
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	20220428	20220428	20220428
crypto-policies-scripts	20220428		
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	2,127	2,127	
cyrus-sasl-plain	2,127		
dbus	1,12,28	1,12,28	

Package	AMI	AMI minimale	Conteneur
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	
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	

Package	AMI	AMI minimale	Conteneur
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	
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

Package	AMI	AMI minimale	Conteneur
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
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

Package	AMI	AMI minimale	Conteneur
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		
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 (aarch64)	2,06 (aarch64)	
grub2-efi-x64-ec2	2,06 (x86_64)	2,06 (x86_64)	

Package	AMI	AMI minimale	Conteneur
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		
hwdata	0,353	0,353	
info	6.7		
inih	49	49	
initscripts	10,09	10,09	
iproute	5.10.0	5.10.0	
iputils	20210202	20210202	

Package	AMI	AMI minimale	Conteneur
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	0,14
kbd	2.4.0	2.4.0	
kbd-misc	2.4.0	2.4.0	
kernel	6,1,90	6,1,90	
kernel-livepatch-repo-s3	2023,4.20240513	2023,4.20240513	
kernel-srpm-macros	1.0		
kernel-tools	6,1,90		
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	1,21	1,21
less	608	608	

Package	AMI	AMI minimale	Conteneur
libacl	2.3.1	2.3.1	2.3.1
libaio	0,3,111		
libarchive	3.5.3	3.5.3	3.5.3
libargon2	27/12/2017	27/12/2017	
libassuan	2.5.5	2.5.5	2.5.5
libattr	2.5.1	2.5.1	2.5.1
libbasicobjects	0,11		
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		
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	

Package	AMI	AMI minimale	Conteneur
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
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	0,13		
libmnl	1.0.4	1.0.4	
libmodulemd	2.13.0	2.13.0	2.13.0

Package	AMI	AMI minimale	Conteneur
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	0,15		
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	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

Package	AMI	AMI minimale	Conteneur
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
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		

Package	AMI	AMI minimale	Conteneur
libverto	0,3.2	0,3.2	0,3.2
libverto-libev	0,3.2		
libxcrypt	4.4.33	4.4.33	4.4.33
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
lm_sensors-libs	3.6.0		
lmdb-libs	0,9,29		
logrotate	3.20.1	3.20.1	
lsof	4,94,0		
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

Package	AMI	AMI minimale	Conteneur
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		
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	

Package	AMI	AMI minimale	Conteneur
openssh	8,7 p1	8,7 p1	
openssh-clients	8,7 p1	8,7 p1	
openssh-server	8,7 p1	8,7 p1	
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	1,77	1,77	
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		
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	10,40	10,40	10,40
pcre2-syntax	10,40	10,40	10,40
perl-Carp	1,50		
perl-Class-Struct	0,66		

Package	AMI	AMI minimale	Conteneur
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		
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		

Package	AMI	AMI minimale	Conteneur
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		
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		

Package	AMI	AMI minimale	Conteneur
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	2021,0726		
perl-Time-Local	1,300		
perl-vars	1,05		
pkgconf	1.8.0		
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		

Package	AMI	AMI minimale	Conteneur
psacct	6.6.4		
psmisc	23,4	23,4	
publicsuffix-list-dafsa	20240212	20240212	20240212
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	
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	

Package	AMI	AMI minimale	Conteneur
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	
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	

Package	AMI	AMI minimale	Conteneur
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	0,1,6	0,1,6	
python3-oauthlib	3.0.2	3.0.2	
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	

Package	AMI	AMI minimale	Conteneur
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	
python3-ruamel-yaml-clib	0,12	0,12	
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		

Package	AMI	AMI minimale	Conteneur
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
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

Package	AMI	AMI minimale	Conteneur
selinux-policy	37,22	37,22	
selinux-policy-targeted	37,22	37,22	
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		
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	252,16	252,16	
systemd-libs	252,16	252,16	
systemd-networkd	252,16	252,16	
systemd-pam	252,16	252,16	

Package	AMI	AMI minimale	Conteneur
systemd-resolved	252,16	252,16	
systemd-udev	252,16	252,16	
system-release	2023,4.20240513	2023,4.20240513	2023,4.20240513
systemtap-runtime	4.8		
tar	1,34	1,34	
tbb	2020,3		
tcpdump	4,99,1		
tcsh	6,24,07		
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		

Package	AMI	AMI minimale	Conteneur
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		
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	

Utilisation de l'AL2023 dans des conteneurs

> **Note**
>
> Pour plus d'informations sur l'utilisation de l'AL2023 pour héberger des charges de travail conteneurisées sur Amazon ECS, consultez. AL2023 pour les hôtes de conteneurs Amazon ECS

L'AL2023 peut être utilisé de plusieurs manières à l'intérieur des conteneurs en fonction du cas d'utilisation. Elle Image de conteneur de base AL2023 est très similaire à une image de conteneur Amazon Linux 2 et à l'AMI minimale AL2023.

Pour les utilisateurs avancés, nous proposons une image de conteneur minimale, introduite dans la version AL2023.2, ainsi qu'une documentation décrivant comment créer des conteneurs simples.

AL2023 peut également être utilisé pour héberger des charges de travail conteneurisées, d'images de conteneurs basées sur AL2023 ou de conteneurs basés sur d'autres distributions Linux. Vous pouvez utiliser AL2023 pour les hôtes de conteneurs Amazon ECS ou utiliser directement les packages d'exécution de conteneurs fournis. Les packages `docker`, `containerd` et `nerdctl` peuvent être installés et utilisés sur AL2023.

Rubriques

Utilisation de l'image du conteneur de base AL2023

L'image du conteneur AL2023 est créée à partir des mêmes composants logiciels que ceux inclus dans l'AMI AL2023. Elle peut être utilisée dans tout environnement en tant qu'image de base pour les charges de travail Docker. Si vous utilisez l'AMI Amazon Linux pour des applications dans Amazon Elastic Compute Cloud (Amazon EC2), vous pouvez conteneuriser vos applications à l'aide de l'image de conteneur Amazon Linux.

Utilisez l'image du conteneur Amazon Linux dans votre environnement de développement local, puis envoyez votre application à AWS Amazon Elastic Container Service (Amazon ECS). Pour plus d'informations, consultez Utilisation d'images Amazon ECR avec Amazon ECS dans le Guide de l'utilisateur Amazon Elastic Container Registry.

L'image de conteneur Amazon Linux est disponible sur Amazon ECR Public. Vous pouvez faire part de vos commentaires pour AL2023 par l'intermédiaire de votre AWS représentant désigné ou en signalant un problème dans le référentiel amazon-linux-2023 sur. GitHub

Pour extraire l'image de conteneur Amazon Linux depuis Amazon ECR Public

1. Authentifiez votre client Docker dans le registre public Amazon Linux. Les jetons d'authentification sont valides pendant 12 heures. Pour plus d'informations, consultez Authentification de registre privé dans le Guide de l'utilisateur Amazon Elastic Container Registry.

   > **Note**
   >
   > La get-login-password commande est prise en charge à l'aide de la dernière version de AWS CLI la version 2. Pour plus d'informations, consultez Installation d' AWS Command Line Interface dans le Guide de l'utilisateur AWS Command Line Interface .

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

   La sortie est la suivante.

   ```
   Login succeeded
   ```

2. Procédez à l'extraction de l'image de conteneur Amazon Linux en exécutant la commande docker pull. Pour afficher l'image du conteneur Amazon Linux dans la galerie publique Amazon ECR, consultez Galerie publique Amazon ECR – amazonlinux.

   > **Note**
   >
   > Lorsque vous extrayez l'image de conteneur Docker AL2023, vous pouvez utiliser les balises dans l'un des formats suivants :

- Pour obtenir la dernière version de l'image de conteneur AL2023, utilisez la balise `:2023`.
- Pour obtenir une version spécifique d'AL2023, vous pouvez utiliser le format suivant :
  - `:2023.[0-7 release quarter].[release date].[build number]`

Les exemples suivants utilisent la balise `:2023` et extraient l'image de conteneur la plus récente disponible d'AL2023.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023
```

3. (Facultatif) Exécutez le conteneur localement.

```
$ docker run -it --security-opt seccomp=unconfined public.ecr.aws/amazonlinux/
amazonlinux:2023 /bin/bash
```

Pour extraire l'image de conteneur AL2023 de Docker Hub

1. Procédez à l'extraction de l'image de conteneur AL2023 à l'aide de la commande docker pull.

```
$ docker pull amazonlinux:2023
```

2. (Facultatif) Exécutez le conteneur localement.

```
$ docker run -it amazonlinux:2023 /bin/bash
```

> **Note**
>
> L'image de conteneur d'AL2023 utilise uniquement le gestionnaire de packages `dnf` pour installer les packages logiciels. Cela signifie qu'il n'existe aucun `amazon-linux-extras` ni aucune commande équivalente à utiliser pour les logiciels supplémentaires.

AL2023 Image minimale du conteneur

> **Note**
>
> Les images de conteneur AL2023 standard conviennent à la plupart des cas d'utilisation, et l'adaptation à l'image de conteneur minimale est susceptible de demander plus de travail que l'adaptation à l'image de conteneur de base AL2023.

L'image de conteneur minimal AL2023, introduite dans AL2023.2, diffère de l'image de conteneur de base car elle ne contient que le strict minimum de packages nécessaires pour installer d'autres packages. L'image de conteneur minimale est conçue pour être un ensemble minimal de packages, et non un ensemble de packages pratique.

L'image de conteneur minimale AL2023 est créée à partir de composants logiciels déjà disponibles dans AL2023. La principale différence entre l'image minimale du conteneur est de l'utiliser `microdnf` pour fournir le gestionnaire de `dnf` packages plutôt que l'image Python basée sur l'ensemble des fonctionnalités`dnf`. Cela permet de réduire la taille de l'image de conteneur minimale tout en évitant de disposer de l'ensemble des fonctionnalités du gestionnaire de `dnf` packages inclus dans les AMI AL2023 et l'image du conteneur de base.

L'image de conteneur minimale AL2023 constitue la base de l'environnement d'exécution `provided.al2023` AWS Lambda.

Pour une liste détaillée des packages inclus dans l'image du conteneur minimal, voirComparaison des packages installés sur les images de conteneurs Amazon Linux 2023.

Image de l'image de conteneur minimale

Comme l'image du conteneur minimal AL2023 contient moins de packages que l'image du conteneur de base de l'AL2023, elle est également nettement plus petite. Le tableau suivant compare les options d'image de conteneur des versions actuelles et passées d'Amazon Linux.

> **Note**
>
> La taille de l'image est celle indiquée dans Amazon Linux sur la galerie publique Amazon ECR.

Image	Version	Taille de l'image	Remarque
Amazon Linux 1 (AL1)	2018,03.0.20230918,0	62,3 Mo	`x86-64` uniquement
Amazon Linux 2	2,0,20230926,0	64,2 Mo	La taille d'`aarch64` est supérieure de 1,6 Mo à celle de `x86-64`
Image de conteneur de base Amazon Linux 2023	2023,2.2023 1002,0	52,4 Mo	
Image de conteneur minimale Amazon Linux 2023	2023.2.20231002.0-minimal	35,2 Mo	

Utilisation de l'image de conteneur minimale AL2023

L'image de conteneur minimal AL2023 est disponible sur ECR et la `2023-minimal` balise pointe toujours vers la dernière image de conteneur minimale basée sur AL2023, tandis que la `minimal` balise peut être mise à jour vers une version d'Amazon Linux plus récente que AL2023.

Vous pouvez extraire ces balises à l'`docker`aide de l'exemple suivant :

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:minimal
```

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
```

L'exemple suivant montre un `Dockerfile` qui prend l'image minimale du conteneur et installe GCC dessus :

```
FROM public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
RUN dnf install -y gcc && dnf clean all
```

Construire des images de conteneurs AL2023 rudimentaires

L'image du conteneur AL2023 est créée à partir des mêmes composants logiciels que ceux inclus dans l'AMI AL2023. Il inclut un logiciel qui permet à la couche conteneur de base de se comporter de la même manière qu'une exécution sur une instance Amazon EC2, tel que le gestionnaire de packages. `dnf` Cette section explique comment créer un conteneur à partir de zéro qui inclut uniquement le strict minimum de dépendances nécessaires à une application.

> **Note**
>
> Les images de conteneur AL2023 standard conviennent à la plupart des cas d'utilisation. L'utilisation de l'image de conteneur standard facilite la création par-dessus votre image. Une image de conteneur simplifiée complique la création à partir de votre image.

Pour créer un conteneur avec le strict minimum de dépendances pour une application

1. Déterminez les dépendances de votre environnement d'exécution. Elles varient en fonction de votre application.
2. Construisez un `Dockerfile` / `Containerfile` qui génère `FROM scratch`. L'exemple suivant de `Dockerfile` peut être utilisé pour générer un conteneur contenant uniquement le shell `bash` et ses dépendances.

```
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"]
```

   - Ce `Dockerfile` fonctionne comme suit :

1. Il démarre un conteneur AL2023 nommé `build`. Ce conteneur sera utilisé pour amorcer le conteneur élémentaire. Il n'est pas déployé lui-même, mais génère le conteneur à déployer.
2. Il crée le répertoire `/sysroot`. Ce répertoire sera l'emplacement où le conteneur `build` installera les dépendances nécessaires au conteneur élémentaire. Dans une étape ultérieure, le chemin `/sysroot` sera empaqueté comme répertoire racine de notre image élémentaire.

   L'utilisation de l'option `--installroot` pour `dnf` de cette manière permet de créer les autres images AL2023. Il s'agit d'une fonctionnalité de `dnf` qui permet aux programmes d'installation et aux outils de création d'images de fonctionner.
3. Il invoque `dnf` pour installer des packages dans `/sysroot`.

   La commande `rpm -q system-release --qf '%{VERSION}'` interroge (`-q`) le package `system-release`, en définissant le format de requête (`--qf`) pour fournir en sortie la version du package interrogé (la variable `%{VERSION}` est la variable `rpm` correspondant à la version du RPM).

   En définissant l'argument `--releasever` de `dnf` sur la version de `system-release` dans le conteneur `build`, ce `Dockerfile` peut être utilisé pour reconstruire le conteneur élémentaire chaque fois qu'une image de base de conteneur mise à jour d'Amazon Linux est publiée.

   Il est possible de définir n'importe quelle version `--releasever` d'Amazon Linux 2023, telle que 2023.4.20240513. Cela signifierait que le `build` conteneur fonctionnerait sous la dernière version d'AL2023, mais que le conteneur barebones serait créé à partir du 2023.4.20240513, quelle que soit la version actuelle d'AL2023.

   L'option de configuration `--setopt=install_weak_deps=False` indique à `dnf` d'installer uniquement les dépendances requises plutôt que celles recommandées ou suggérées.
4. Il copie le système installé à la racine d'un conteneur vierge (`FROM scratch`).
5. Il définit `ENTRYPOINT` comme binaire souhaité, dans ce cas `/bin/bash`.

3. Créez un répertoire vide et ajoutez le contenu de l'exemple à l'étape 2 dans un fichier nommé `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. Générez le conteneur en exécutant la commande suivante.

```
$ docker build -t al2023-barebones-bash-example
```

5. Exécutez le conteneur à l'aide de la commande suivante pour voir à quel point un conteneur dédié à `bash` est minimal.

```
$ 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
```

Pour un exemple plus pratique, la procédure suivante crée un conteneur pour une application C qui affiche `Hello World!`.

1. Créez un répertoire vide et ajoutez le code source C et `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. Générez le conteneur à l'aide de la commande suivante.

```
$ docker build -t al2023-barebones-c-hello-world-example .
```

3. Exécutez le conteneur à l'aide de la commande suivante.

```
$ docker run -it --rm al2023-barebones-c-hello-world-example
```

```
Hello World!
```

Comparaison des packages installés sur les images de conteneurs Amazon Linux 2023

Comparaison des RPM présents sur l'image du conteneur de base AL2023 par rapport aux RPM présents sur l'image minimale du conteneur AL2023.

Package	Conteneur	Récipient minimal
alternatives	1.15	1.15
amazon-linux-repo-cdn	2023,4.20240513	2023,4.20240513
audit-libs	3,0.6	3,0.6
basesystem	11	11
bash	5.2,15	5.2,15
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023,2,64	2023,2,64
coreutils-single	8,32	8,32
crypto-policies	20220428	20220428
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	

Package	Conteneur	Récipient minimal
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
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	1,21

Package	Conteneur	Récipient minimal
libacl	2.3.1	2.3.1
libarchive	3.5.3	3.5.3
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
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

Package	Conteneur	Récipient minimal
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
libuuid	2,37,4	2,37,4
libverto	0,3.2	0,3.2
libxcrypt	4.4.33	
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

Package	Conteneur	Récipient minimal
microdnf		3.8.1
microdnf-dnf		3.8.1
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	10,40	10,40
pcre2-syntax	10,40	10,40
popt	1,18	1,18
publicsuffix-list-dafsa	20240212	20240212
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	

Package	Conteneur	Récipient minimal
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
system-release	2023,4.20240513	2023,4.20240513
tzdata	2024a	
xz-libs	5.2.5	5.2.5
yum	4.14.0	
zlib	1.2.11	1.2.11

Comparaison des packages installés sur les images de conteneurs et l'AMI minimale Amazon Linux 2023

Comparaison des RPM présents sur l'AMI minimale AL2023 avec les RPM présents sur la base AL2023 et les images de conteneur minimales.

Package	AMI minimale	Conteneur	Récipient minimal
alternatives	1.15	1.15	1.15
amazon-chrony-config	4.3		
amazon-ec2-net-utils	2.4.1		
amazon-linux-repo-cdn		2023,4.20240513	2023,4.20240513
amazon-linux-repo-s3	2023,4.20240513		
amazon-linux-sb-keys	2023.1		
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	2023,2,64	2023,2,64	2023,2,64
checkpolicy	3.4		
chrony	4.3		
cloud-init	22.2.2		
cloud-init-cfg-ec2	22.2.2		

Package	AMI minimale	Conteneur	Récipient minimal
cloud-utils-growpart	0,31		
coreutils	8,32		
coreutils-common	8,32		
coreutils-single		8,32	8,32
cpio	2,13		
cracklib	2,9,6		
cracklib-dicts	2,9,6		
crypto-policies	20220428	20220428	20220428
cryptsetup-libs	2.6.1		
curl-minimal	8.5.0	8.5.0	8.5.0
cyrus-sasl-lib	2,127		
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		

Package	AMI minimale	Conteneur	Récipient minimal
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		
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	

Package	AMI minimale	Conteneur	Récipient minimal
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
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		

Package	AMI minimale	Conteneur	Récipient minimal
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
grep	3.8	3.8	3.8
groff-base	1.22.4		
grub2-common	2,06		
grub2-efi-aa64-ec2	2,06 (aarch64)		
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		

Package	AMI minimale	Conteneur	Récipient minimal
hwdata	0,353		
inih	49		
initscripts	10,09		
iproute	5.10.0		
iputils	20210202		
irqbalance	1.9.0		
jansson	2.14		
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	6,1,90		
kernel-livepatch-repo-s3	2023,4.20240513		
keyutils-libs	1.6.3	1.6.3	1.6.3
kmod	29		
kmod-libs	29		
krb5-libs	1,21	1,21	1,21
less	608		

Package	AMI minimale	Conteneur	Récipient minimal
libacl	2.3.1	2.3.1	2.3.1
libarchive	3.5.3	3.5.3	3.5.3
libargon2	27/12/2017		
libassuan	2.5.5	2.5.5	2.5.5
libattr	2.5.1	2.5.1	2.5.1
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		
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

Package	AMI minimale	Conteneur	Récipient minimal
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		
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

Package	AMI minimale	Conteneur	Récipient minimal
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
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	4.4.33	4.4.33	
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
logrotate	3.20.1		

Package	AMI minimale	Conteneur	Récipient minimal
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)		
microdnf			3.8.1
microdnf-dnf			3.8.1
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	8,7 p1		
openssh-clients	8,7 p1		
openssh-server	8,7 p1		
openssl	3,0.8		

Package	AMI minimale	Conteneur	Récipient minimal
openssl-libs	3,0.8	3,0.8	3,0.8
openssl-pkcs11	0,4,12		
os-prober	1,77		
p11-kit	0,24.1	0,24.1	0,24.1
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	10,40	10,40	10,40
pcre2-syntax	10,40	10,40	10,40
policycoreutils	3.4		
popt	1,18	1,18	1,18
procps-ng	3.3,17		
psmisc	23,4		
publicsuffix-list-dafsa	20240212	20240212	20240212
python3	3,9,16	3,9,16	
python3-attrs	20.3.0		
python3-audit	3,0.6		
python3-awscrt	0,19,19		

Package	AMI minimale	Conteneur	Récipient minimal
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	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		

Package	AMI minimale	Conteneur	Récipient minimal
python3-jmespath	0.10.0		
python3-jsonpatch	1,21		
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	0,1,6		
python3-oauthlib	3.0.2		
python3-pip-wheel	21.3.1	21.3.1	
python3-ply	3,11		

Package	AMI minimale	Conteneur	Récipient minimal
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	4.16.1.3	
python3-ruamel-yaml	0,16.6		
python3-ruamel-yaml-clib	0,12		
python3-setools	4.4.1		

Package	AMI minimale	Conteneur	Récipient minimal
python3-setuptools	59,6,0		
python3-setuptools-wheel	59,6,0	59,6,0	
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

Package	AMI minimale	Conteneur	Récipient minimal
selinux-policy	37,22		
selinux-policy-targeted	37,22		
setup	2.13.7	2.13.7	2.13.7
shadow-utils	4,9		
sqlite-libs	3,40,0	3,40,0	3,40,0
sudo	1,9,15		
sysctl-defaults	1.0		
systemd	252,16		
systemd-libs	252,16		
systemd-networkd	252,16		
systemd-pam	252,16		
systemd-resolved	252,16		
systemd-udev	252,16		
system-release	2023,4.20240513	2023,4.20240513	2023,4.20240513
tar	1,34		
tzdata	2024a	2024a	
update-motd	2.2		
userspace-rcu	0.12.1		

Package	AMI minimale	Conteneur	Récipient minimal
util-linux	2,37,4		
util-linux-core	2,37,4		
vim-data	9,0.2153		
vim-minimal	9,0.2153		
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 activé AWS Elastic Beanstalk

AWS Elastic Beanstalk est un service de déploiement et de mise à l'échelle d'applications et de services Web. Chargez votre code pour qu'Elastic Beanstalk gère automatiquement le déploiement, du provisionnement de la capacité à l'équilibrage de charge, en passant par l'autoscaling et la surveillance de l'état de l'application. Pour plus d'informations, consultez AWS Elastic Beanstalk.

Pour utiliser Elastic Beanstalk, vous devez créer une application, charger une version d'application sous la forme d'un bundle de fichiers source d'application (par exemple, un fichier Java .war) sur Elastic Beanstalk, puis fournir des informations sur l'application. Elastic Beanstalk lance

automatiquement un environnement, crée et AWS configure les ressources nécessaires à l'exécution de votre code. Pour plus d'informations, consultez le Guide du développeur AWS Elastic Beanstalk.

Les plateformes Linux Elastic Beanstalk utilisent des instances Amazon EC2 et ces instances exécutent Amazon Linux. Depuis le 4 août 2023, Elastic Beanstalk propose les branches de plateforme suivantes basées sur Amazon Linux 2023 : Docker, Tomcat, Java SE, Node.js, PHP et Python. Elastic Beanstalk vise à élargir la prise en charge d'AL2023 à d'autres plateformes Elastic Beanstalk.

La liste complète des plateformes Elastic Beanstalk prises en charge et des plateformes Elastic Beanstalk actuelles créées par-dessus AL2023 se trouve dans la section Plateformes Linux Elastic Beanstalk du Guide du développeur Elastic Beanstalk.

Vous trouverez les notes de mise à jour des nouvelles plateformes Elastic Beanstalk et les versions des plateformes existantes dans les notes de mise à jour d'Elastic Beanstalk.

Utilisation d'AL2023 dans AWS CloudShell

AWS CloudShell est un shell pré-authentifié basé sur un navigateur que vous pouvez lancer directement depuis le. AWS Management Console Vous pouvez y accéder CloudShell AWS Management Console de différentes manières. Pour plus d'informations, consultez la section Comment démarrer avec AWS CloudShell ?

AWS CloudShell, qui est actuellement basé sur Amazon Linux 2, migrera vers AL2023. La migration vers AL2023 commencera à être déployée dans l'ensemble Régions AWS à partir du 4 décembre 2023. Pour plus d'informations sur CloudShell la migration vers AL2023, consultez la section AWS CloudShell Migration d'Amazon Linux 2 vers Amazon Linux 2023.

Utilisation d'AMI Amazon ECS basées sur AL2023 pour héberger des charges de travail conteneurisées

> **Note**
>
> Pour plus d'informations sur l'utilisation de l'AL2023 à l'intérieur d'un conteneur, voirAL2023 dans des conteneurs.

Amazon Elastic Container Service (Amazon ECS) est un service d'orchestration de conteneurs entièrement géré qui vous permet de déployer, de gérer et de dimensionner aisément des applications conteneurisées. En tant que service entièrement géré, Amazon ECS intègre les meilleures pratiques opérationnelles et de AWS configuration. Il est intégré à la fois à des outils AWS et à des outils tiers, tels qu'Amazon Elastic Container Registry (Amazon ECR) et Docker. Cette intégration permet aux équipes de se concentrer plus facilement sur la création des applications, et non sur l'environnement. Vous pouvez exécuter et mettre à l'échelle vos charges de travail de conteneurs entre régions AWS dans le cloud, sans avoir à gérer un plan de contrôle.

Vous pouvez héberger des charges de travail conteneurisées sur AL2023 à l'aide de l'AMI optimisée pour Amazon ECS basée sur AL2023. Pour plus d'informations, consultez l'AMI optimisée pour Amazon ECS

Modifications apportées à la norme AL2023 pour Amazon ECS par rapport à la version AL2

Comme AL2, AL2023 fournit les packages de base requis pour fonctionner en tant qu'instance Linux Amazon ECS. Dans AL2`containerd`, les `ecs-init` packages`docker`, et étaient disponibles via`amazon-linux-extras`, tandis que AL2023 inclut ces packages dans les référentiels principaux.

Grâce à la fonctionnalité de mise à niveau déterministe via des référentiels versionnés, chaque AMI AL2023 est verrouillée par défaut sur une version de référentiel spécifique. Cela est également vrai pour l'AMI optimisée pour Amazon ECS pour AL2023. Toutes les mises à jour de votre environnement peuvent être gérées et testées avec soin avant le déploiement, ce qui permet de revenir facilement au contenu d'une ancienne AMI en cas de problème. Pour plus d'informations sur cette fonctionnalité AL2023, consultez Utilisation de mises à niveau déterministes via un référentiel versionné sur AL2023.

AL2023 passe à cgroup v2 via l'interface cgroup v1 prise en charge dans AL2. Pour plus d'informations, consultez Hiérarchie des groupes de contrôle unifiés (cgroup v2).

> **Note**
>
> Les versions AL2023 antérieures à 2023.2.20230920 (la première version d'AL2023.2) contenaient un bogue `systemd` pour la gestion des pertes de mémoire (OOM) au sein d'un cgroup. Tous les processus du cgroup étaient toujours supprimés au lieu que le tueur OOM choisisse un processus à la fois, ce qui est le comportement prévu.

Il s'agissait d'une régression par rapport au comportement d'AL2, et elle est corrigée depuis la version 2023.2.20230920 d'AL2023.

Le code permettant de créer l'AMI optimisée pour Amazon ECS est disponible dans le amazon-ecs-ami GitHub projet. Les notes de publication décrivent quelle version d'AL2023 correspond à quelle version de l'AMI Amazon ECS.

Personnalisation de l'AMI optimisée pour Amazon ECS basée sur AL2023

Important

Nous vous recommandons d'utiliser l'AMI AL2023 optimisée pour Amazon ECS. Pour plus d'informations, consultez l'AMI optimisée pour Amazon ECS dans le manuel Amazon Elastic Container Service Developer Guide.

Vous pouvez utiliser les mêmes scripts de génération qu'Amazon ECS pour créer des AMI personnalisées. Pour plus d'informations, consultez le script de génération de l'AMI Linux optimisé pour Amazon ECS.

Utilisation d'Amazon Elastic File System sur AL2023

Amazon Elastic File System (Amazon EFS) fournit un stockage de fichiers entièrement élastique sans serveur pour vous permettre de partager des données de fichiers sans provisionner ni gérer la capacité et les performances de stockage. Amazon EFS est conçu pour se mettre à l'échelle à la demande et peut atteindre plusieurs pétaoctets sans perturber les applications. Il augmente ou diminue automatiquement la capacité au fil de vos ajouts et suppressions de fichiers. Comme Amazon EFS propose une interface de services web simple, vous pouvez créer et configurer des systèmes de fichiers rapidement et facilement. Le service gère toute l'infrastructure de stockage de fichiers pour vous, ce qui vous libère des tâches compliquées liées au déploiement, à l'application de correctifs et à la gestion des configurations de systèmes de fichiers complexes.

Amazon EFS prend en charge le protocole Network File System version 4 (NFSv4.1 et NFSv4.0), afin que les applications et outils que vous utilisez aujourd'hui fonctionnent en toute transparence avec Amazon EFS. Plusieurs instances de calcul, notamment Amazon EC2, Amazon ECS et Amazon AWS Lambda, peuvent accéder à un système de fichiers Amazon EFS en même temps.

Par conséquent, un système de fichiers EFS peut fournir une source de données commune pour les charges de travail et les applications exécutées sur plusieurs instances de calcul ou serveurs.

Installation d'**`amazon-efs-utils`** sur AL2023

Le `amazon-efs-utils` package est disponible dans les référentiels AL2023 pour être installé et utilisé pour accéder aux systèmes de fichiers Amazon EFS.

Installation du package **`amazon-efs-utils`** sur AL2023

- Effectuez `amazon-efs-utils` l'installation à l'aide de la commande suivante.

```
$ dnf -y install amazon-efs-utils
```

Montage d'un système de fichiers Amazon EFS sur AL2023

Une fois `amazon-efs-utils` installé, vous pouvez monter un système de fichiers Amazon EFS sur votre instance AL2023.

Monter un système de fichiers Amazon EFS sur AL2023

- Pour effectuer un montage à l'aide de l'identifiant du système de fichiers, utilisez la commande suivante.

```
sudo mount -t efs file-system-id efs-mount-point/
```

Vous pouvez également monter le système de fichiers de manière à ce que les données en transit soient chiffrées à l'aide du protocole TLS, ou en utilisant le nom DNS ou l'adresse IP cible du montage au lieu de l'ID du système de fichiers. Pour plus d'informations, consultez Montage sur des instances Amazon Linux à l'aide de l'assistant de montage EFS.

Utilisation d'Amazon EMR basé sur AL2023

Amazon EMR est un service web qui facilite le traitement efficace de grandes quantités de données à l'aide d'Apache Hadoop et des services proposés par AWS.

Publications Amazon EMR basées sur AL2023

La version 7.0.0 d'Amazon EMR était la première version basée sur AL2023. Avec cette version, AL2023 est le système d'exploitation de base d'Amazon EMR, apportant tous les avantages d'AL2023 à Amazon EMR. Pour plus d'informations, consultez les notes de mise à jour d'Amazon EMR 7.0.0.

AL2023 basé sur Amazon EMR sur EKS

Amazon EMR sur EKS 6.13 a été la première version à introduire AL2023 en option. Avec cette version, vous pouvez lancer Spark avec AL2023 comme système d'exploitation, ainsi que l'environnement d'exploitation Java 17. Pour plus d'informations, consultez les notes de mise à jour d'Amazon EMR sur EKS 6.13 et toutes les notes de mise à jour d'Amazon EMR sur EKS.

Utilisation d'AL2023 dans AWS Lambda

Avec AWS Lambda, vous pouvez exécuter du code sans provisionner ni gérer de serveurs. Vous payez uniquement le temps de calcul que vous consommez ; aucun frais ne s'applique quand votre code ne s'exécute pas. Vous pouvez exécuter du code pour quasiment n'importe quel type d'application ou de service backend, sans aucune administration. Il vous suffit de télécharger votre code et Lambda s'occupe de tout ce qui est nécessaire à l'exécution de votre code et à sa mise à l'échelle en garantissant une haute disponibilité.

Environnement d'exécution **`provided.al2023`** géré et image de conteneur AL2023

Le runtime `provided.al2023` de base est basé sur l'image de conteneur minimale AL2023 et fournit un environnement d'exécution géré Lambda basé sur AL2023 et une image de base de conteneur. Le temps `provided.al2023` d'exécution étant basé sur l'image de conteneur minimale AL2023, il est nettement inférieur à 40 Mo par rapport au temps `provided.al2` d'exécution d'environ 109 Mo.

Pour plus d'informations, consultez les sections Runtimes Lambda et Utilisation d'images de conteneurs Lambda.

Runtimes Lambda basés sur AL2023

Les futures versions des environnements d'exécution en langage géré, telles que Node.js 20, Python 3.12, Java 21 et .NET 8, sont basées sur AL2023 et seront utilisées `provided.al2023` comme image de base, comme décrit dans l'annonce des environnements d'exécution basés sur AL2023.

Fonctions Lambda basées sur AL2023

- Fonctions Lambda AL2023 écrites en Go
- Fonctions Lambda AL2023 écrites en Rust

Pour en savoir plus amples, consultez la section Quotas Lambda du Guide du développeur AWS Lambda .

Didacticiels

Les didacticiels suivants vous montrent comment effectuer des tâches courantes à l'aide d'instances Amazon EC2 exécutant Amazon Linux 2023 (AL2023). Pour les didacticiels vidéo, voir Vidéos AWS pédagogiques et ateliers.

Pour les instructions AL2, consultez les didacticiels pour les instances Amazon EC2 exécutant Linux dans le guide de l'utilisateur Amazon EC2.

Didacticiels

- Tutoriel : Installation d'un serveur LAMP sur AL2023
- Tutoriel : Configurer SSL/TLS sur AL2023
- Tutoriel : Héberger un WordPress blog sur AL2023

Tutoriel : Installation d'un serveur LAMP sur AL2023

Les procédures suivantes vous aident à installer un serveur Web Apache compatible avec PHP et MariaDB (un fork de MySQL développé par la communauté) sur votre instance AL2023 (parfois appelée serveur Web LAMP ou stack LAMP). Vous pouvez utiliser ce serveur pour héberger un site web statique ou déployer une application PHP dynamique qui lit et écrit des informations sur une base de données.

> **Important**
>
> Ces procédures sont destinées à être utilisées avec AL2023. Si vous essayez de configurer un serveur Web LAMP sur une autre distribution, comme Ubuntu ou Red Hat Enterprise Linux, ce tutoriel ne fonctionnera pas. Pour Ubuntu, consultez la documentation de la communauté Ubuntu suivante : ApacheMySQLPHP. Pour les autres distributions, consultez leur documentation spécifique.

Tâches

- Étape 1 : Préparer le serveur LAMP
- Étape 2 : Tester votre serveur LAMP
- Étape 3 : Sécuriser le serveur de base de données

Étape 1 : Préparer le serveur LAMP

Prérequis

- Ce didacticiel part du principe que vous avez déjà lancé une nouvelle instance à l'aide d'AL2023, avec un nom DNS public accessible depuis Internet. Pour plus d'informations, consultez AL2023 sur Amazon EC2. Vous devez aussi avoir configuré votre groupe de sécurité pour permettre les connexions SSH (port 22), HTTP (port 80) et HTTPS (port 443). Pour plus d'informations sur ces prérequis, consultez Autoriser le trafic entrant pour vos instances Linux dans le guide de l'utilisateur Amazon EC2.
- La procédure suivante installe la dernière version de PHP disponible sur AL2023, actuellement 8.1. Si vous prévoyez d'utiliser d'autres applications PHP que celles décrites dans ce tutoriel, vous pouvez vérifier qu'elles sont compatibles avec 8.1.

Pour préparer le serveur LAMP

1. Connectez-vous à votre instance. Pour plus d'informations, consultez Connexion aux instances AL2023.
2. Pour vous assurer que tous vos packages logiciels sont mis à jour, effectuez une mise à jour logicielle rapide sur votre instance. Ce processus peut prendre quelques minutes, mais il est important pour vous assurer que vous disposez des dernières mises à jour de sécurité et des nouveaux correctifs de bogues.

   L'option `-y` installe les mises à jour sans demander de confirmation. Si vous souhaitez examiner les mises à jour avant l'installation, vous pouvez omettre cette option.

   ```
   [ec2-user ~]$ sudo dnf update -y
   ```

3. Installez les dernières versions du serveur Web Apache et des packages PHP pour AL2023.

   ```
   [ec2-user ~]$ sudo dnf install -y httpd wget php-fpm php-mysqli php-json php php-devel
   ```

4. Installez le serveur web MariaDB et les packages logiciels. Utilisez la commande dnf install pour installer plusieurs packages logiciels et toutes les dépendances associées au même moment.

```
[ec2-user ~]$ sudo dnf install mariadb105-server
```

Vous pouvez afficher les versions actuelles de ces packages avec la commande suivante :

```
[ec2-user ~]$ sudo dnf info package_name
```

Exemple :

```
[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. Démarrez le serveur web Apache.

```
[ec2-user ~]$ sudo systemctl start httpd
```

6. Utilisez la commande systemctl pour configurer le serveur Web Apache afin qu'il soit lancé à chaque démarrage système.

```
[ec2-user ~]$ sudo systemctl enable httpd
```

Vous pouvez vérifier que httpd est activé en exécutant la commande suivante :

```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

7. Ajoutez une règle de sécurité pour autoriser les connexions HTTP entrantes (port 80) à votre instance si vous ne l'avez pas déjà fait. Par défaut, un groupe de sécurité launch-wizard-*N* a été créé pour votre instance lors du lancement. Si vous n'avez pas ajouté de règle de groupe de sécurité supplémentaire, ce groupe contient une règle unique pour autoriser les connexions SSH.

   a. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.

   b. Dans le navigateur de gauche, choisissez Instances, puis sélectionnez votre instance.

   c. Sous l'onglet Sécurité, affichez les règles entrantes. Vous devriez voir la règle suivante :

```
Port range   Protocol     Source
22           tcp          0.0.0.0/0
```

> **Warning**
>
> L'utilisation de `0.0.0.0/0` permet à toutes les adresses IPv4 d'accéder à votre instance à l'aide du protocole SSH. Cette solution est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. Dans un environnement de production, vous autorisez uniquement l'accès à votre instance pour une adresse IP ou une plage d'adresses spécifiques.

   d. S'il n'existe aucune règle entrante autorisant les connexions HTTP (port 80), vous devez ajouter la règle maintenant. Choisissez le lien pour le groupe de sécurité. À l'aide des procédures décrites dans la section Autoriser le trafic entrant pour vos instances Linux, ajoutez une nouvelle règle de sécurité entrante avec les valeurs suivantes :

      - Type : HTTP

- Protocole : TCP
- Plage de ports : 80
- Source : Personnalisé

8. Testez votre serveur web. Dans un navigateur web, saisissez l'adresse DNS publique (ou l'adresse IP publique) de votre instance. S'il n'y a pas de contenu dans `/var/www/html`, vous devriez voir la page de test d'Apache, qui affichera le message « Ça marche ! ».

   Vous pouvez obtenir le DNS public de votre instance à l'aide de la console Amazon EC2 (vérifiez la colonne Public IPv4 DNS (DNS IPv4 public) ; si cette colonne est masquée, choisissez l'icône Preferences (Préférences) (icône en forme d'engrenage) et activez Public IPv4 DNS (DNS IPv4 public)).

   Vérifiez que le groupe de sécurité de l'instance contient une règle autorisant le trafic HTTP sur le port 80. Pour plus d'informations, voir Ajouter des règles au groupe de sécurité.

   > **Important**
   >
   > Si vous n'utilisez pas Amazon Linux, il se peut que vous deviez aussi configurer le pare-feu sur votre instance pour autoriser ces connexions. Pour obtenir plus d'informations sur la configuration du pare-feu, consultez la documentation de votre distribution spécifique.

La commande httpd traite les fichiers qui sont conservés dans un répertoire appelé racine du document Apache. La racine du document Apache d'Amazon Linux est `/var/www/html` qui est détenu par défaut par la racine.

Pour autoriser le compte `ec2-user` à manipuler les fichiers de ce répertoire, vous devez modifier la propriété et les autorisations du répertoire. Il existe plusieurs façons d'accomplir cette tâche. Dans ce didacticiel, vous ajoutez l'utilisateur `ec2-user` au groupe `apache` pour donner au groupe `apache` la propriété du répertoire `/var/www` et attribuer les autorisations d'écriture au groupe.

Pour définir les autorisations sur les fichiers

1. Ajoutez votre utilisateur (dans ce cas, `ec2-user`) au groupe `apache`.

```
[ec2-user ~]$ sudo usermod -a -G apache ec2-user
```

2. Déconnectez-vous, puis reconnectez-vous pour sélectionner le nouveau groupe, puis vérifiez votre adhésion.

   a. Déconnectez-vous (utilisez la commande exit ou fermez la fenêtre de terminal) :

   ```
   [ec2-user ~]$ exit
   ```

   b. Pour vérifier votre adhésion au groupe `apache`, reconnectez-vous à votre instance, puis exécutez la commande suivante :

   ```
   [ec2-user ~]$ groups
   ec2-user adm wheel apache systemd-journal
   ```

3. Remplacez la propriété de groupe de `/var/www` et son contenu par le groupe `apache`.

   ```
   [ec2-user ~]$ sudo chown -R ec2-user:apache /var/www
   ```

4. Pour ajouter des autorisations d'écriture de groupe et définir l'ID de groupe pour les futurs sous-répertoires, modifiez les autorisations sur les répertoires de `/var/www` et ses sous-répertoires.

   ```
   [ec2-user ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod
    2775 {} \;
   ```

5. Pour ajouter des autorisations d'écriture de groupe, modifiez de façon récursive les autorisations sur les fichiers de `/var/www` et ses sous-répertoires :

   ```
   [ec2-user ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
   ```

Maintenant, `ec2-user` (et tous les futurs membres du groupe `apache`) peut ajouter, supprimer et modifier les fichiers à la racine du document Apache. Vous pouvez ainsi ajouter du contenu, tel qu'un site Web statique ou une application PHP.

Pour sécuriser votre serveur web (facultatif)

Un serveur web exécutant le protocole HTTP ne fournit aucune sécurité de transport pour les données qu'il envoie ou reçoit. Lorsque vous vous connectez à un serveur HTTP via un navigateur Web, les URL que vous visitez, le contenu des pages web que vous recevez et le contenu (y compris les mots de passe) de tous les formulaires HTML que vous envoyez peuvent être vus par des personnes malveillantes sur le chemin d'accès réseau. Les bonnes pratiques en matière de

sécurisation de votre serveur web consistent à installer la prise en charge HTTPS (HTTP Secure), qui protège vos données grâce au chiffrement SSL/TLS.

Pour plus d'informations sur l'activation de HTTPS sur votre serveur, consultez Tutoriel : Configurer SSL/TLS sur AL2023.

Étape 2 : Tester votre serveur LAMP

Si votre serveur est installé et en cours d'exécution, et que vos autorisations sur les fichiers sont correctement définies, votre compte `ec2-user` doit pouvoir créer un fichier PHP simple dans le répertoire `/var/www/html` qui est disponible à partir d'Internet.

Pour tester votre serveur LAMP

1. Créez un fichier PHP à la racine du document Apache.

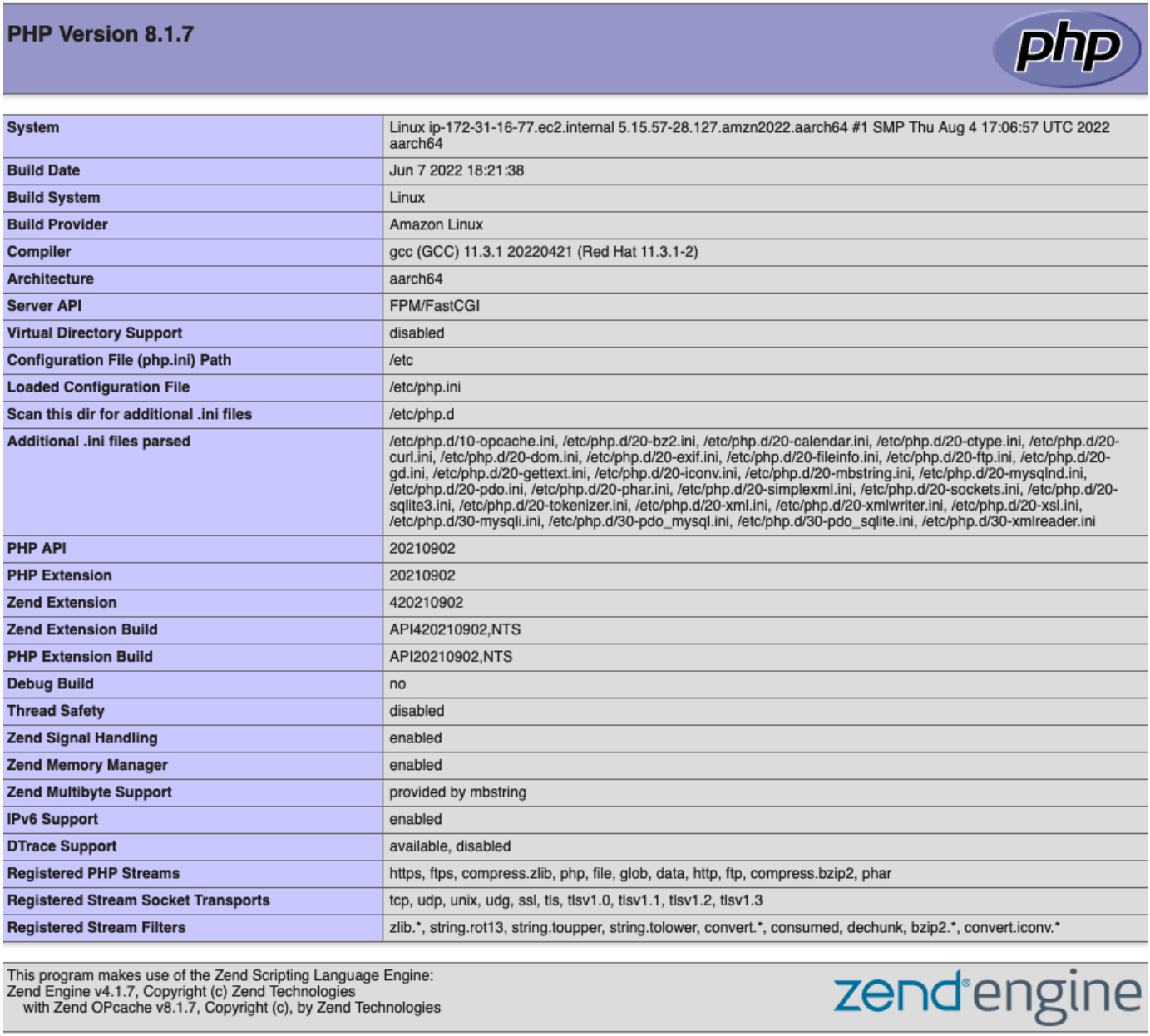
   ```
   [ec2-user ~]$ echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
   ```

   Si l'erreur « Permission denied » s'affiche lorsque vous essayez d'exécuter cette commande, essayez de vous déconnecter et de vous reconnecter pour récupérer les autorisations d'un groupe que vous avez configurées dans Pour définir les autorisations sur les fichiers.

2. Dans un navigateur web, saisissez l'URL du fichier que vous venez de créer. Cette URL est l'adresse DNS publique de votre instance suivie par une barre oblique et le nom du fichier. Exemples :

   ```
   http://my.public.dns.amazonaws.com/phpinfo.php
   ```

   Vous devriez voir la page d'informations 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

Si vous ne voyez pas cette page, vérifiez que le fichier `/var/www/html/phpinfo.php` a été créé correctement à l'étape précédente. Vous pouvez également vérifier que les packages requis ont été installés avec la commande suivante.

```
[ec2-user ~]$ sudo dnf list installed httpd mariadb-server php-mysqlnd
```

Si l'un des packages requis n'est pas présent dans votre sortie, installez-les avec la commande sudo yum install *package*.

3. Supprimez le fichier `phpinfo.php`. Même si ces informations peuvent vous être utiles, elles ne doivent pas être diffusées sur Internet pour des raisons de sécurité.

```
[ec2-user ~]$ rm /var/www/html/phpinfo.php
```

Vous devriez maintenant avoir un serveur web LAMP entièrement fonctionnel. Si vous ajoutez un contenu à la racine du document Apache à l'emplacement `/var/www/html`, vous devez pouvoir voir ce contenu à l'adresse du DNS public de votre instance.

Étape 3 : Sécuriser le serveur de base de données

L'installation par défaut du serveur MariaDB possède plusieurs fonctions qui sont parfaites pour les tests et le développement, mais elles devraient être désactivées ou supprimées des serveurs de production. La commande mysql_secure_installation vous guide à travers le processus de paramétrage d'un mot de passe racine et de suppression des fonctions non sécurisées de votre installation. Même si vous ne comptez pas utiliser le serveur MariaDB, nous vous recommandons de suivre cette procédure.

Pour sécuriser le serveur MariaDB

1. Démarrez le serveur MariaDB.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

2. Exécutez mysql_secure_installation.

```
[ec2-user ~]$ sudo mysql_secure_installation
```

   a. A l'invite, saisissez un mot de passe pour le compte racine.

      i. Saisissez le mot de passe racine actuel. Par défaut, le compte racine n'a pas de mot de passe défini. Appuyez sur Entrée.

      ii. Tapez **Y** pour définir un mot de passe et saisissez deux fois un mot de passe sécurisé. Pour plus d'informations sur la création d'un mot de passe fiable, consultez https://identitysafe.norton.com/password-generator/. Assurez-vous de stocker ce mot de passe en lieu sûr.

La mesure la plus simple pour sécuriser votre base de données consiste à définir un mot de passe racine pour MariaDB. Lorsque vous concevez ou installez une application reposant sur une base de données, vous devez généralement créer un utilisateur de services de base de données pour cette application et éviter d'utiliser le compte racine, sauf pour administrer la base de données.

b. Tapez **Y** pour supprimer les comptes d'utilisateur anonymes.

c. Tapez **Y** pour désactiver la connexion racine à distance.

d. Tapez **Y** pour supprimer la base de données de test.

e. Tapez **Y** pour recharger les tableaux de privilèges et enregistrer vos changements.

3. (Facultatif) Si vous ne comptez pas utiliser le serveur MariaDB tout de suite, arrêtez-le. Vous pouvez le redémarrer lorsque vous en avez de nouveau besoin.

```
[ec2-user ~]$ sudo systemctl stop mariadb
```

4. (Facultatif) Si vous voulez que le serveur MariaDB soit lancé à chaque démarrage, saisissez la commande suivante.

```
[ec2-user ~]$ sudo systemctl enable mariadb
```

Étape 4 : (facultatif) Installation phpMyAdmin

phpMyAdminest un outil de gestion de base de données basé sur le Web que vous pouvez utiliser pour afficher et modifier les bases de données MySQL sur votre instance EC2. Suivez les étapes ci-dessous pour installer et configurer `phpMyAdmin` sur votre instance Amazon Linux.

> **Important**
>
> Nous ne vous recommandons pas d'utiliser `phpMyAdmin` pour accéder à un serveur LAMP, sauf si vous avez activé SSL/TLS dans Apache. Sinon, votre mot de passe administrateur de base de données et d'autres données sont transmises de façon non sécurisée sur Internet. Pour connaître les recommandations de sécurité des développeurs, consultez la section Sécurisation de votre phpMyAdmin installation. Pour obtenir des informations générales sur la sécurisation d'un serveur Web sur une instance EC2, consultez Tutoriel : Configurer SSL/TLS sur AL2023.

Pour installer phpMyAdmin

1. Installez les dépendances obligatoires.

```
[ec2-user ~]$ sudo dnf install php-mbstring php-xml -y
```

2. Redémarrez Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

3. Redémarrez `php-fpm`.

```
[ec2-user ~]$ sudo systemctl restart php-fpm
```

4. Accédez à la racine du document Apache sur `/var/www/html`.

```
[ec2-user ~]$ cd /var/www/html
```

5. Sélectionnez un package source pour la dernière phpMyAdmin version sur https://www.phpmyadmin.net/downloads. Pour télécharger le fichier directement sur votre instance, copiez le lien et collez-le dans une commande wget, comme dans cet exemple :

```
[ec2-user html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
```

6. Créez un dossier `phpMyAdmin` et extrayez le package dans celui-ci avec la commande suivante.

```
[ec2-user html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1
```

7. Supprimez l'archive *phpMyAdmin- latest-all-languages .tar.gz*.

```
[ec2-user html]$ rm phpMyAdmin-latest-all-languages.tar.gz
```

8. (Facultatif) Si le serveur MySQL n'est pas en cours d'exécution, démarrez-le maintenant.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

9. Dans un navigateur Web, saisissez l'URL de votre phpMyAdmin installation. Cette URL est l'adresse DNS publique (ou l'adresse IP publique) de votre instance suivie par une barre oblique et le nom du fichier de votre répertoire d'installation. Par exemple :

```
http://my.public.dns.amazonaws.com/phpMyAdmin
```

Vous devriez voir la page phpMyAdmin de connexion :

10. Connectez-vous à votre phpMyAdmin installation avec le nom d'`root`utilisateur et le mot de passe root MySQL que vous avez créés précédemment.

    Votre installation doit être configurée avant que vous la mettiez en service. Nous vous suggérons de commencer par créer manuellement le fichier de configuration, comme suit :

    a. Pour commencer avec un fichier de configuration minimal, utilisez votre éditeur de texte favori pour créer un nouveau fichier, puis copiez le contenu de `config.sample.inc.php` dans celui-ci.

b. Enregistrez le fichier `config.inc.php` dans le phpMyAdmin répertoire qui le contient`index.php`.

c. Reportez-vous aux instructions après la création du fichier dans la section Utilisation du script d' phpMyAdmin installation des instructions d'installation pour toute configuration supplémentaire.

Pour plus d'informations sur l'utilisation phpMyAdmin, consultez le guide de phpMyAdmin l'utilisateur.

Dépannage

Cette section propose des suggestions pour résoudre les problèmes courants que vous pouvez rencontrer lors de la configuration d'un nouveau serveur LAMP.

Je ne parviens pas à me connecter à mon serveur à l'aide d'un navigateur Web.

Effectuez les vérifications suivantes pour voir si votre serveur web Apache est en cours d'exécution et accessible.

- Le serveur web est-il en cours d'exécution ?

  Vous pouvez vérifier que httpd est activé en exécutant la commande suivante :

  ```
  [ec2-user ~]$ sudo systemctl is-enabled httpd
  ```

  Si le processus httpd n'est pas en cours d'exécution, répétez les étapes décrites dans Pour préparer le serveur LAMP.

- Le pare-feu est-il configuré correctement ?

  Vérifiez que le groupe de sécurité de l'instance contient une règle autorisant le trafic HTTP sur le port 80. Pour plus d'informations, voir Ajouter des règles au groupe de sécurité.

Je ne parviens pas à me connecter à mon serveur en utilisant HTTPS

Effectuez les vérifications suivantes pour voir si votre serveur Web Apache est configuré pour prendre en charge HTTPS.

- Le serveur Web est-il correctement configuré ?

  Après avoir installé Apache, le serveur est configuré pour le trafic HTTP. Pour prendre en charge HTTPS, activez TLS sur le serveur et installez un certificat SSL. Pour plus d'informations, veuillez consulter Tutoriel : Configurer SSL/TLS sur AL2023.
- Le pare-feu est-il configuré correctement ?

  Vérifiez que le groupe de sécurité de l'instance contient une règle autorisant le trafic HTTPS sur le port 443. Pour plus d'informations, consultez Autoriser le trafic entrant pour vos instances Linux.

Rubriques en relation

Pour plus d'informations sur le transfert de fichiers vers votre instance ou l'installation d'un WordPress blog sur votre serveur Web, consultez la documentation suivante :

- Transférez des fichiers vers votre instance Linux à l'aide de WinSCP dans le guide de l'utilisateur Amazon EC2.
- Transférez des fichiers vers des instances Linux à l'aide d'un client SCP dans le guide de l'utilisateur Amazon EC2.
- Tutoriel : Héberger un WordPress blog sur AL2023

Pour plus d'informations sur les commandes et le logiciel utilisés dans ce tutoriel, consultez les pages web suivantes :

- Serveur Web Apache : http://httpd.apache.org/
- Serveur de base de données MariaDB : https://mariadb.org/
- Langage de programmation PHP : http://php.net/

Pour plus d'informations sur l'enregistrement d'un nom de domaine pour votre serveur web ou le transfert d'un nom de domaine existant vers cet hôte, consultez Création et migration de domaines et de sous-domaines vers Amazon Route 53 dans le Amazon Route 53 Manuel du développeur.

Tutoriel : Configurer SSL/TLS sur AL2023

SSL/TLS (Secure Sockets Layer/Transport Layer Security) crée un canal chiffré entre un serveur web et un client web qui empêche les données en transit d'être écoutées. Ce didacticiel explique

comment ajouter manuellement la prise en charge de SSL/TLS sur une instance EC2 avec AL2023 et un serveur Web Apache. Ce tutoriel suppose que vous n'utilisez pas d'équilibreur de charge. Si vous utilisez Elastic Load Balancing, vous pouvez choisir de configurer le déchargement SSL sur l'équilibreur de charge, en utilisant un certificat à partir de AWS Certificate Manager.

Pour des raisons historiques, le chiffrement web est communément appelé SSL. Alors que les navigateurs web prennent toujours en charge SSL, son protocole successeur TLS est moins vulnérable en cas d'attaque. AL2023 désactive le support côté serveur pour toutes les versions de SSL par défaut. Les organismes de normalisation de la sécurité considèrent que TLS 1.0 n'est pas sûr. TLS 1.0 et TLS 1.1 sont devenus officiellement obsolètes en mars 2021. Ce tutoriel contient des conseils pour l'activation de TLS 1.2 exclusivement. Le protocole TLS 1.3 a été finalisé en 2018 et est disponible en AL2 tant que la bibliothèque TLS sous-jacente (OpenSSL dans ce didacticiel) est prise en charge et activée. Les clients doivent prendre en charge le protocole TLS 1.2 ou une version ultérieure d'ici le 28 juin 2023. Pour plus d'informations sur les normes de chiffrement mises à jour, consultez RFC 7568 et RFC 8446.

Ce tutoriel fait référence au chiffrement Web moderne simplement comme TLS.

Important

Ces procédures sont destinées à être utilisées avec AL2023. Si vous essayez de configurer une instance EC2 exécutant une distribution différente ou une instance exécutant une ancienne version de Amazon Linux, certaines procédures de ce tutoriel peuvent ne pas fonctionner. Pour Ubuntu, consultez la documentation de la communauté Ubuntu suivante : Open SSL on Ubuntu (Ouvrir SSL sur Ubuntu). Pour Red Hat Enterprise Linux, consultez les informations suivantes :Setting up the Apache HTTP Web Server (Configuration du serveur web HTTP Apache). Pour les autres distributions, consultez leur documentation spécifique.

Note

Vous pouvez également utiliser AWS Certificate Manager (ACM) for AWS Nitro enclaves, une application d'enclave qui vous permet d'utiliser des certificats SSL/TLS publics et privés avec vos applications Web et vos serveurs exécutés sur des instances Amazon EC2 avec Nitro Enclaves. AWS Nitro Enclaves est une fonctionnalité d'Amazon EC2 qui permet de créer des environnements de calcul isolés pour protéger et traiter en toute sécurité des données très sensibles, telles que des certificats SSL/TLS et des clés privées.

ACM for Nitro Enclaves fonctionne avec nginx exécuté sur votre instance Amazon EC2 Linux pour créer des clés privées, distribuer des certificats et des clés privées et gérer le renouvellement des certificats.
Pour utiliser ACM for Nitro Enclaves, vous devez utiliser une instance Linux compatible avec les enclaves.
Pour plus d'informations, consultez Qu'est-ce que AWS Nitro Enclaves ? et AWS Certificate Manager pour Nitro Enclaves dans le guide de l'utilisateur de AWS Nitro Enclaves.

Table des matières

Prérequis

Avant de commencer ce tutoriel, suivez les étapes suivantes :

- Lancez une instance AL2023 basée sur EBS. Pour plus d'informations, consultez AL2023 sur Amazon EC2.
- Configurez vos groupes de sécurité afin que votre instance puisse accepter des connexions sur les ports TCP suivants :
  - SSH (port 22)
  - HTTP (port 80)
  - HTTPS (port 443)

  Pour plus d'informations, consultez Autoriser le trafic entrant pour vos instances Linux dans le guide de l'utilisateur Amazon EC2.
- Installez le serveur Web Apache. Pour step-by-step obtenir des instructions, voirTutoriel : Installation d'un serveur LAMP sur AL2023. Seuls le package httpd et ses dépendances sont nécessaires. Par conséquent, vous pouvez ignorer les instructions impliquant PHP et MariaDB.
- Pour identifier et authentifier les sites web, l'infrastructure à clés publiques (PKI) TLS repose sur le système de noms de domaine (DNS). Pour utiliser votre instance EC2 pour héberger un site web

public, vous devez enregistrer un nom de domaine pour votre serveur web ou transférer un nom de domaine existant vers votre hôte Amazon EC2. Plusieurs services d'enregistrement de domaines tiers et d'hébergement DNS sont disponibles pour cela, ou vous pouvez utiliser Amazon Route 53.

Étape 1 : Activer TLS sur le serveur

Cette procédure vous guide tout au long du processus de configuration du protocole TLS sur AL2023 avec un certificat numérique autosigné.

> **Note**
>
> Un certificat auto-signé est acceptable dans un environnement de test, mais pas pour les environnements de production. Si vous exposez votre certificat auto-signé sur Internet, les visiteurs de votre site verront s'afficher des messages d'avertissement de sécurité.

Pour activer TLS sur un serveur

1. Connectez-vous à votre instance et confirmez qu'Apache est en cours d'exécution. Pour plus d'informations, consultez Connexion aux instances AL2023.

   ```
   [ec2-user ~]$ sudo systemctl is-enabled httpd
   ```

   Si la valeur renvoyée n'est pas « activé », démarrez Apache et configurez-le pour qu'il démarre à chaque amorçage du système.

   ```
   [ec2-user ~]$ sudo systemctl start httpd && sudo systemctl enable httpd
   ```

2. Pour vous assurer que tous vos packages logiciels sont mis à jour, effectuez une mise à jour logicielle rapide sur votre instance. Ce processus peut prendre quelques minutes, mais il est important pour vous assurer que vous disposez des dernières mises à jour de sécurité et des nouveaux correctifs de bogues.

   > **Note**
   >
   > L'option -y installe les mises à jour sans demander de confirmation. Si vous souhaitez examiner les mises à jour avant l'installation, vous pouvez omettre cette option.

```
[ec2-user ~]$ sudo dnf install openssl mod_ssl
```

3. Une fois que vous avez saisi la commande suivante, vous êtes redirigé vers une invite où vous pouvez saisir des informations sur votre 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
```

Cela génère un nouveau fichier `apache-selfsigned.crt` dans le répertoire `/etc/pki/tls/certs/`. Le nom de fichier spécifié correspond au nom par défaut attribué dans la directive SSLCertificateFile dans `/etc/httpd/conf.d/ssl.conf`.

Votre instance dispose désormais des fichiers suivants que vous utilisez pour configurer votre serveur sécurisé et créer un certificat pour les tests :

- `/etc/httpd/conf.d/ssl.conf`

  Le fichier de configuration de mod_ssl. Il contient des directives indiquant à Apache où trouver les clés et les certificats de chiffrement, les versions de protocoles TLS à autoriser et les algorithmes de chiffrement à accepter. Voici votre fichier de certificat local :

- `/etc/pki/tls/certs/apache-selfsigned.crt`

Ce fichier contient un certificat auto-signé et la clé privée du certificat. Apache exige que le certificat et la clé soient au format PEM qui est constitué de caractères ASCII codés en Base64, encadrés par des lignes « BEGIN » et « END », comme dans l'exemple abrégé ci-après.

```
-----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-----
```

Les noms et extensions de fichiers sont fournis à titre indicatif et n'ont aucun effet sur la fonction. Par exemple, vous pouvez appeler un certificat `cert.crt`, `cert.pem`, ou tout autre nom de fichier dans la mesure où la directive associée dans le fichier `ssl.conf` utilise le même nom.

> **Note**
>
> Lorsque vous remplacez les fichiers TLS par défaut par vos propres fichiers personnalisés, veillez à ce qu'ils soient au format PEM.

4. Redémarrez Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

> **Note**
>
> Assurez-vous que le port TCP 443 est accessible sur votre instance EC2, tel que décrit précédemment.

5. Votre serveur web Apache devrait maintenant prendre en charge HTTPS (HTTP sécurisé) sur le port 443. Testez-le en saisissant l'adresse IP ou le nom de domaine complet de votre instance EC2 dans une barre URL du navigateur avec le préfixe **`https://`**.

   Étant donné que vous vous connectez à un site avec un certificat d'hôte auto-signé non approuvé, il se peut que votre navigateur affiche une série d'avertissements de sécurité. Ignorez-les et poursuivez sur le site.

Si la page de test Apache par défaut s'ouvre, cela signifie que vous avez configuré correctement TLS sur votre serveur. Toutes les données transmises entre le navigateur et le serveur sont maintenant chiffrées.

> **Note**
>
> Pour éviter aux visiteurs du site d'avoir des avertissements, vous devez obtenir un certificat signé par une CA qui chiffre mais vous authentifie aussi publiquement comme le propriétaire du site.

Étape 2 : Obtenir un certificat signé par une autorité de certification (CA)

Vous pouvez utiliser le processus suivant pour obtenir un certificat signé par une CA :

- Générez une demande de signature de certificat (CSR) à partir d'une clé privée
- Envoyez la demande de signature de certificat (CSR) à une autorité de certification (CA)
- Obtenez un certificat d'hôte signé
- Configurez Apache pour utiliser le certificat

Le chiffrement d'un certificat X.509 TLS auto-signé est identique à celui d'un certificat signé par une autorité de certification. La différence est sociale, pas mathématique. Une autorité de certification promet, au minimum, de valider la propriété d'un domaine avant de générer un certificat pour un demandeur. Chaque navigateur web contient une liste d'autorités de certification approuvées par le fournisseur de navigateur pour faire cela. Un certificat X.509 se compose surtout d'une clé publique qui correspond à votre clé de serveur privée et d'une signature de l'autorité de certification qui est cryptographiquement reliée à la clé publique. Lorsqu'un navigateur se connecte à un serveur web sur HTTPS, le serveur présente un certificat que le navigateur doit vérifier par rapport à sa liste d'autorités de certification approuvées. Si le signataire est sur la liste ou s'il est accessible via une chaîne de confiance composée d'autres utilisateurs de confiance, le navigateur négocie un canal de données chiffrées rapide avec le serveur et charge la page.

Les certificats coûtent généralement de l'argent à cause travail impliqué dans la validation des requêtes, donc il est intéressant de comparer les prix. Quelques autorités de certification offrent des certificats basiques gratuits. La plus importante autorité de certification est le projet Let's Encrypt, qui prend également en charge l'automatisation du processus de création et de renouvellement des

certificats. Pour plus d'informations sur l'utilisation d'un certificat Let's Encrypt, veuillez consulter la rubrique Get Certbot.

Si vous prévoyez d'offrir des services de qualité commerciale, AWS Certificate Manager est une bonne option.

La clé est l'élément sous-jacent du certificat d'hôte. Depuis 2019, des groupes gouvernementaux et industriels recommandent l'utilisation d'une taille de clé minimale (module) de 2048 bits pour les clés RSA conçues pour protéger des documents, jusqu'en 2030. La taille du module par défaut générée par OpenSSL dans AL2023 est de 2048 bits, ce qui convient à une utilisation dans un certificat signé par une autorité de certification. Dans la procédure suivante, étape facultative pour ceux qui souhaitent une clé personnalisée, par exemple, une clé avec un module plus important ou utilisant un algorithme de chiffrement différent.

> **Important**
>
> Ces instructions pour l'acquisition de certificats d'hôte signés par l'autorité de certification (CA) ne fonctionnent pas à moins que vous possédiez un domaine DNS enregistré et hébergé.

Pour obtenir un certificat signé par une CA

1. Connectez-vous à votre instance et accédez à /etc/pki/tls/private/. Il s'agit du répertoire où vous stockez la clé privée du serveur pour TLS. Si vous préférez utiliser une clé d'hôte existante pour générez la CSR, passez à l'étape 3. Pour plus d'informations sur la connexion à votre instance, voir Connexion aux instances AL2023
2. (Facultatif) Générez une nouvelle clé privée. Voici quelques exemples de configurations de clés. Toutes les clés obtenues fonctionnent avec votre serveur web, mais elles diffèrent dans le degré et le type de sécurité qu'elles mettent en œuvre.

   - Exemple 1 : création d'une clé d'hôte RSA par défaut. Le fichier obtenu, **`custom.key`**, est une clé privée RSA 2048 bits.

     ```
     [ec2-user ~]$ sudo openssl genrsa -out custom.key
     ```

   - Exemple 2 : création d'une clé RSA plus forte avec un modulus plus grand. Le fichier obtenu, **`custom.key`**, est une clé privée RSA 4096 bits.

```
[ec2-user ~]$ sudo openssl genrsa -out custom.key 4096
```

- Exemple 3 : création d'une clé RSA chiffrée 4096 bits avec protection par mot de passe. Le fichier obtenu, **`custom.key`**, est une clé privée RSA 4096 bits chiffrée avec le chiffrement AES-128.

  > **Important**
  >
  > Le chiffrement de la clé offre une plus grande sécurité, mais comme une clé chiffrée nécessite un mot de passe, les services qui en dépendent ne peuvent pas démarrer automatiquement. A chaque fois que vous utilisez cette clé, vous devez fournir le mot de passe (« abcde12345 » dans l'exemple précédent) sur une connexion SSH.

```
[ec2-user ~]$ sudo openssl genrsa -aes128 -passout pass:abcde12345 -out
 custom.key 4096
```

- Exemple 4 : création d'une clé avec un chiffrement non RSA. La cryptographie RSA peut être relativement lente en raison de la taille de ses clés publiques, lesquelles sont basées sur le produit de deux grands nombres premiers. Cependant, il est possible de créer des clés pour TLS qui utilisent des chiffrements non RSA. Les clés basées sur les mathématiques des courbes elliptiques sont plus petites et plus rapides en termes de calcul, tout en offrant un niveau de sécurité équivalent.

  ```
  [ec2-user ~]$ sudo openssl ecparam -name prime256v1 -out custom.key -genkey
  ```

  Le résultat est une clé privée 256 bits à courbes elliptiques utilisant prime256v1, une « courbe nommée » que OpenSSL prend en charge. Sa qualité cryptographique est légèrement plus importante qu'une clé RSA 2048 bits, selon NIST.

  > **Note**
  >
  > Toutes les autorités de certification ne fournissent pas le même niveau de support pour elliptic-curve-based les clés que pour les clés RSA.

Assurez-vous que la nouvelle clé privée possède un critère de propriété et d'autorisations très restrictif (propriétaire=racine, groupe=racine, lecture/écriture pour propriétaire uniquement). Les commandes seraient similaires à celles illustrées dans l'exemple suivant.

```
[ec2-user ~]$ sudo chown root:root custom.key
[ec2-user ~]$ sudo chmod 600 custom.key
[ec2-user ~]$ ls -al custom.key
```

Les commandes ci-avant produisent le résultat suivant.

```
-rw------- root root custom.key
```

Une fois que vous avez créé et configuré une clé satisfaisante, vous pouvez créer une CSR.

3. Créez une CSR à l'aide de votre clé préférée. L'exemple suivant utilise **`custom.key`**.

```
[ec2-user ~]$ sudo openssl req -new -key custom.key -out csr.pem
```

OpenSSL ouvre une boîte de dialogue et vous invite à compléter les informations affichées dans le tableau ci-dessous. Tous les champs à l'exception de Common Name sont facultatifs pour un certificat d'hôte basique avec validation de domaine.

Nom	Description	Exemple
Nom du pays	Abréviation ISO de deux lettres de votre pays.	US (=Etats-Unis)
Nom de l'état ou de la province	Nom de l'état ou de la province où votre organisation se situe. Ce nom ne peut pas être abrégé.	Washington
Nom de la localité	L'emplacement de votre organisation, comme une ville.	Seattle
Nom de l'organisation	Nom légal complet de votre organisation. N'abrégez pas le nom de votre organisation.	Exemple d'entreprise

Nom	Description	Exemple
Nom de l'unité d'organisation	Informations supplémentaires sur l'organisation, s'il y en a.	Exemple de service
Nom commun	Cette valeur doit correspondre exactement à l'adresse web que les utilisateurs saisiront dans un navigateur, selon vous. Il s'agit généralement d'un nom de domaine avec un nom d'hôte ou un alias préfixé sous la forme **`www.example.com`** . Dans les essais avec un certificat auto-signé et aucune résolution DNS, le nom commun peut se composer uniquement du nom d'hôte. Les autorités de certification proposent aussi des certificats onéreux qui acceptent les noms inconnus comme **`*.example.com`** .	www.exemple.com
Adresse e-mail	L'adresse e-mail de l'administrateur du serveur.	quelquun@exemple.com

Au final, OpenSSL vous invite à donner un mot de passe de stimulation facultatif. Ce mot de passe s'applique uniquement à la CSR et aux transactions entre vous et votre autorité de certification, donc suivez les recommandations de l'autorité de certification sur cela, l'autre champ facultatif et le nom de l'entreprise facultatif. Le mot de passe de stimulation de la CSR n'a aucun effet sur le fonctionnement du serveur.

Le fichier obtenu **`csr.pem`** contient votre clé publique, la signature numérique de votre clé publique et les métadonnées que vous avez saisies.

4. Envoyez la CSR à une autorité de certification. Elle consiste généralement en l'ouverture de votre fichier CSR dans un éditeur de texte et la reproduction du contenu dans un formulaire web. A ce moment-là, il se peut que l'on vous demande de fournir un SAN (subject alternate name) ou plus à placer sur le certificat. Si **`www.example.com`** est le nom commun, **`example.com`** serait un bon SAN, et vice versa. Un visiteur de votre site qui saisit l'un de ces noms devrait bénéficier

d'une connexion sans erreur. Si le formulaire web de votre autorité de certification le permet, incluez le nom commun dans la liste des SAN. Certaines autorités de certification l'incluent automatiquement.

Une fois que votre demande a été approuvée, vous recevrez un nouveau certificat d'hôte signé par l'autorité de certification. Il se peut que l'on vous demande également de télécharger un fichier de certificat intermédiaire qui contient des certificats supplémentaires nécessaires pour compléter la chaîne de confiance de l'autorité de certification.

> **Note**
>
> Votre autorité de certification peut vous envoyer des fichiers sous différents formats en fonction des finalités recherchées. Dans ce tutoriel, vous devez utiliser uniquement un fichier de certificat au format PEM, qui comporte habituellement (mais pas toujours) une extension de fichier `.pem` ou `.crt`. Si vous ne savez pas quel fichier utiliser, ouvrez les fichiers dans un éditeur de texte et recherchez celui qui contient un ou plusieurs blocs commençant par la ligne suivante.
>
> ```
> - - - - -BEGIN CERTIFICATE - - - - -
> ```
>
> Le fichier doit également se terminer par la ligne suivante.
>
> ```
> - - - -END CERTIFICATE - - - - -
> ```
>
> Vous pouvez également tester le fichier dans la ligne de commande, comme suit.
>
> ```
> [ec2-user certs]$ openssl x509 -in certificate.crt -text
> ```
>
> Vérifiez que ces lignes apparaissent dans le fichier. N'utilisez pas de fichiers se terminant par `.p7b`, `.p7c` ou autres extensions similaires.

5. Placez le nouveau certificat signé par une CA et les certificats intermédiaires dans le répertoire `/etc/pki/tls/certs`.

> **Note**
>
> Il existe plusieurs méthodes pour charger votre nouveau certificat dans votre instance EC2, mais le moyen le plus simple et informatif consiste à ouvrir un éditeur de texte

(vi, nano, Bloc-notes, etc.) sur votre ordinateur local et votre instance, puis à copier et coller le contenu du fichier. Pour effectuer ces opérations sur l'instance EC2, vous devez disposer de privilèges racine [sudo]. Vous voyez ainsi immédiatement s'il existe des problèmes d'autorisation ou de chemin d'accès. Veillez toutefois à ne pas d'ajouter des lignes supplémentaires lors de la copie du contenu, ou à les modifier de quelque façon.

Depuis le `/etc/pki/tls/certs` répertoire, vérifiez que les paramètres de propriété, de groupe et d'autorisation du fichier correspondent aux valeurs par défaut très restrictives de la norme AL2023 (owner=root, group=root, read/write pour le propriétaire uniquement). L'exemple suivant illustre les commandes à utiliser.

```
[ec2-user certs]$ sudo chown root:root custom.crt
[ec2-user certs]$ sudo chmod 600 custom.crt
[ec2-user certs]$ ls -al custom.crt
```

Ces commandes devraient générer le résultat suivant.

```
-rw------- root root custom.crt
```

Les autorisations pour le fichier de certificat intermédiaire sont moins contraignantes (propriétaire=racine, groupe=racine, le propriétaire peut écrire, le groupe peut lire, tout le monde peut lire). L'exemple suivant illustre les commandes à utiliser.

```
[ec2-user certs]$ sudo chown root:root intermediate.crt
[ec2-user certs]$ sudo chmod 644 intermediate.crt
[ec2-user certs]$ ls -al intermediate.crt
```

Ces commandes devraient générer le résultat suivant.

```
-rw-r--r-- root root intermediate.crt
```

6. Placez la clé privée que vous avez utilisée pour créer la CSR dans le répertoire `/etc/pki/tls/private/`.

> **Note**
>
> Il existe plusieurs méthodes pour charger votre clé personnalisée dans votre instance EC2, mais le moyen le plus simple et informatif consiste à ouvrir un éditeur de texte (vi, nano, Bloc-notes, etc.) sur votre ordinateur local et votre instance, puis à copier et coller le contenu du fichier. Pour effectuer ces opérations sur l'instance EC2, vous devez disposer de privilèges racine [sudo]. Vous voyez ainsi immédiatement s'il existe des problèmes d'autorisation ou de chemin d'accès. Veillez toutefois à ne pas d'ajouter des lignes supplémentaires lors de la copie du contenu, ou à les modifier de quelque façon.

Depuis le `/etc/pki/tls/private` répertoire, utilisez les commandes suivantes pour vérifier que les paramètres de propriété, de groupe et d'autorisation du fichier correspondent aux valeurs par défaut très restrictives de la norme AL2023 (owner=root, group=root, read/write pour le propriétaire uniquement).

```
[ec2-user private]$ sudo chown root:root custom.key
[ec2-user private]$ sudo chmod 600 custom.key
[ec2-user private]$ ls -al custom.key
```

Ces commandes devraient générer le résultat suivant.

```
-rw------- root root custom.key
```

7. Modifiez `/etc/httpd/conf.d/ssl.conf` pour refléter les nouveaux fichiers de certificat et de clé.

   a. Fournissez le chemin et le nom de fichier du certificat d'hôte signé par une CA dans la directive `SSLCertificateFile` d'Apache :

      ```
      SSLCertificateFile /etc/pki/tls/certs/custom.crt
      ```

   b. Si vous avez reçu un fichier de certificat intermédiaire (`intermediate.crt` dans cet exemple), indiquez son nom correct de chemin et de fichier à l'aide de la directive `SSLCACertificateFile` d'Apache :

      ```
      SSLCACertificateFile /etc/pki/tls/certs/intermediate.crt
      ```

> **Note**
>
> Certaines autorités de certification combinent le certificat d'hôte et les certificats intermédiaires dans un seul fichier ; la directive `SSLCACertificateFile` devient alors inutile. Consultez les instructions fournies par votre autorité de certification.

c. Fournissez le chemin et le nom de fichier de la clé privée (`custom.key` dans cet exemple) dans la directive `SSLCertificateKeyFile` d'Apache :

```
SSLCertificateKeyFile /etc/pki/tls/private/custom.key
```

8. Enregistrez `/etc/httpd/conf.d/ssl.conf` et redémarrez Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

9. Testez votre serveur en saisissant votre nom de domaine dans la barre d'URL de navigateur avec le préfixe `https://`. Votre navigateur doit charger la page de test via HTTPS sans générer d'erreurs.

Étape 3 : Tester et renforcer la configuration de sécurité

Une fois que votre TLS est opérationnel et exposé au public, vous devriez tester son niveau de sécurité. Il est facile de le faire avec des services en ligne comme Qualys SSL Labs qui effectue une analyse gratuite et complète de votre configuration de sécurité. En fonction des résultats, vous pouvez décider de renforcer la configuration de sécurité par défaut en contrôlant les protocoles que vous acceptez, les chiffrements que vous préférez et que vous excluez. Pour plus d'informations, consultez comment Qualys formule ses scores.

> **Important**
>
> Le test concret est essentiel pour la sécurité de votre serveur. Les petites erreurs de configuration peuvent entraîner des failles de sécurité et des pertes de données. Comme les pratiques de sécurité recommandées changent constamment en réponse à la recherche et aux menaces émergentes, des audits de sécurité périodiques sont essentiels pour la bonne administration du serveur.

Sur le site Qualys SSL Labs, saisissez le nom de domaine complet de votre serveur dans le formulaire **`www.example.com`**. Après environ deux minutes, vous recevrez une note (de A à F) pour votre site et une analyse détaillée des résultats. Le tableau suivant résume le rapport pour un domaine avec des paramètres identiques à la configuration Apache par défaut sur AL2023, et avec un certificat Certbot par défaut.

Score général	B
Certificat	100 %
Support du protocole	95 %
Échange de clés	70 %
Force du chiffrement	90 %

Même si l'aperçu montre que la configuration est principalement sûre, le rapport détaillé indique plusieurs problèmes potentiels, répertoriés ici dans l'ordre de gravité :

✗ Le chiffrement RC4 est pris en charge pour être utilisé par certains navigateurs plus anciens. Un chiffrement est le noyau mathématique d'un algorithme de chiffrement. RC4, un chiffrement rapide utilisé pour chiffrer les flux de données TLS, est connu pour avoir plusieurs failles importantes. À moins que vous ayez une très bonne raison de prendre en charge des navigateurs existants, vous devez désactiver cette option.

✗ Les anciennes versions de TLS sont prises en charge. La configuration prend en charge TLS 1.0 (déjà obsolète) et TLS 1.1 (bientôt obsolète). Seul TLS 1.2 est recommandé depuis 2018.

✗ La confidentialité persistante n'est pas entièrement prise en charge. La confidentialité persistante est une fonction des algorithmes qui chiffrent à l'aide de clés de session temporaires (éphémères) issues de la clé privée. Ceci signifie en pratique que les pirates informatiques ne peuvent pas déchiffrer les données HTTPS même s'ils possèdent la clé privée à long terme d'un serveur web.

Pour corriger et prévenir les erreurs de configuration TLS

1. Ouvrez le fichier de configuration `/etc/httpd/conf.d/ssl.conf` dans un éditeur de texte et mettez en commentaire la ligne suivante en saisissant « # » au début de la ligne.

   ```
   #SSLProtocol all -SSLv3
   ```

2. Ajoutez la directive suivante :

```
#SSLProtocol all -SSLv3
SSLProtocol -SSLv2 -SSLv3 -TLSv1 -TLSv1.1 +TLSv1.2
```

Cette directive désactive explicitement les versions SSL 2 et 3, ainsi que les versions TLS 1.0 et 1.1. Le serveur refuse désormais d'accepter les connexions chiffrées avec des clients utilisant tout sauf TLS 1.2. La formulation des commentaires dans la directive indique plus clairement, à un lecteur humain, ce pour quoi le serveur est configuré.

> **Note**
>
> La désactivation des versions TLS 1.0 et 1.1 de cette manière empêche un faible pourcentage de navigateurs web obsolètes d'accéder à votre site.

Pour modifier la liste des chiffrements autorisés

1. Dans le fichier de configuration `/etc/httpd/conf.d/ssl.conf`, recherchez la section avec la directive **`SSLCipherSuite`** et mettez en commentaire la ligne existante en saisissant « # » au début de la ligne.

```
#SSLCipherSuite HIGH:MEDIUM:!aNULL:!MD5
```

2. Spécifiez explicitement des suites de chiffrement et un ordre de chiffrement qui donnent la priorité à la confidentialité persistante et évitent les chiffrements peu sûrs. La directive `SSLCipherSuite` utilisée ici est basée sur la sortie du Mozilla SSL Configuration Generator, qui adapte une configuration TLS au logiciel spécifique s'exécutant sur votre serveur. (Pour plus d'informations, consultez la ressource utile de Mozilla Security/Server Side TLS.) Déterminez d'abord vos versions d'Apache et OpenSSL en utilisant la sortie des commandes suivantes.

```
[ec2-user ~]$ yum list installed | grep httpd

[ec2-user ~]$ yum list installed | grep openssl
```

Par exemple, si l'information renvoyée est Apache 2.4.34 et OpenSSL 1.0.2, nous saisissons cela dans le générateur. Si vous choisissez le modèle de compatibilité « moderne », il crée une directive `SSLCipherSuite` qui applique la sécurité de façon stricte, tout en étant compatible

avec la plupart des navigateurs. Si votre logiciel ne prend pas en charge la configuration moderne, vous pouvez mettre à jour le logiciel ou choisir la configuration « intermédiaire ».

```
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
```

Les chiffrements sélectionnés contiennent ECDHE (une abréviation pour Elliptic Curve Diffie-Hellman Ephemeral) dans leur nom. Le terme ephemeral indique la confidentialité persistante. Comme corollaire, ces chiffrements ne prennent pas en charge RC4.

Nous vous recommandons d'utiliser une liste explicite de chiffrements au lieu de compter sur les valeurs par défaut ou les directives succinctes dont le contenu n'est pas visible.

Copiez la directive générée dans `/etc/httpd/conf.d/ssl.conf`.

> **Note**
>
> Même si la directive est affichée ici sur plusieurs lignes afin d'être plus lisible, elle doit être sur une seule ligne lorsqu'elle est copiée dans `/etc/httpd/conf.d/ssl.conf` avec un point (pas d'espace) entre les noms des chiffrements.

3. En dernier lieu, supprimez la mise en commentaire de la ligne suivante en retirant le « # » au début de la ligne.

```
#SSLHonorCipherOrder on
```

Cette directive force le serveur à préférer les chiffrements de niveau élevé notamment (dans ce cas) ceux qui prennent en charge la confidentialité persistante. Avec cette directive activée, le serveur essaie d'établir une connexion très sécurisée avant d'avoir recours aux chiffrements autorisés dotés d'une sécurité moindre.

Après avoir terminé ces deux procédures, enregistrez les modifications dans `/etc/httpd/conf.d/ssl.conf` et redémarrez Apache.

Si vous testez de nouveau le domaine sur Qualys SSL Labs, vous devriez voir que la vulnérabilité RC4 et les autres avertissements ont été supprimés et que le résumé ressemble à ce qui suit.

Score général	A
Certificat	100 %
Support du protocole	100 %
Échange de clés	90 %
Force du chiffrement	90 %

Chaque mise à jour d'OpenSSL présente de nouveaux chiffrements et supprime le support des anciens. Conservez votre instance EC2 AL2023 up-to-date, surveillez les annonces de sécurité d'OpenSSL et soyez attentif aux informations faisant état de nouveaux exploits de sécurité dans la presse technique.

Dépannage

- Mon serveur Web Apache ne démarre pas si je ne fournis pas un mot de passe

  Il s'agit du comportement attendu si vous avez installé une clé de serveur privée chiffrée et protégée par mot de passe.

  Vous pouvez supprimer l'obligation de chiffrement et de mot de passe de la clé. En supposant que vous disposez d'une clé RSA privée chiffrée nommée `custom.key` dans le répertoire par défaut et que le mot de passe de celle-ci est **`abcde12345`**, exécutez les commandes suivantes sur votre instance EC2 pour générer une version non chiffrée de la clé.

```
[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
```

Apache devrait maintenant démarrer sans vous demander de fournir un mot de passe.

- J'obtiens des erreurs lorsque j'exécute sudo dnf install -y mod_ssl.

  Lorsque vous installez les packages requis pour SSL, vous pouvez voir des erreurs similaires à ce qui suit.

  ```
  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
  ```

  Cela signifie généralement que votre instance EC2 n'exécute pas AL2023. Ce didacticiel ne prend en charge que les instances fraîchement créées à partir d'une AMI AL2023 officielle.

Tutoriel : Héberger un WordPress blog sur AL2023

Les procédures suivantes vous aideront à installer, configurer et sécuriser un WordPress blog sur votre instance AL2023. Ce didacticiel est une bonne introduction à l'utilisation d'Amazon EC2 dans la mesure où vous avez le contrôle total d'un serveur Web hébergeant votre WordPress blog, ce qui n'est pas typique d'un service d'hébergement traditionnel.

Vous êtes responsable de la mise à jour des packages logiciels et de la gestion des correctifs de sécurité pour votre serveur. Pour une WordPress installation plus automatisée qui ne nécessite pas d'interaction directe avec la configuration du serveur Web, le AWS CloudFormation service fournit un WordPress modèle qui peut également vous aider à démarrer rapidement. Pour de plus amples informations, veuillez consulter Démarrez dans le AWS CloudFormation Guide de l'utilisateur. Si vous préférez héberger votre WordPress blog sur une instance Windows, consultez la section Déployer un WordPress blog sur votre instance Windows Amazon EC2 dans le guide de l'utilisateur Amazon EC2. Si vous avez besoin d'une solution de haute disponibilité avec une base de données découplée, consultez la section Déploiement d'un WordPress site Web à haute disponibilité dans le guide du développeur.AWS Elastic Beanstalk

> **Important**
>
> Ces procédures sont destinées à être utilisées avec AL2023. Pour obtenir des informations sur d'autres distributions, consultez leur documentation spécifique. Plusieurs étapes de ce tutoriel ne fonctionnent pas sur des instances Ubuntu. Pour obtenir de l'aide concernant l'installation WordPress sur une instance Ubuntu, consultez WordPressla documentation

Ubuntu. Vous pouvez également l'utiliser CodeDeploypour accomplir cette tâche sur les systèmes Amazon Linux, macOS ou Unix.

Rubriques

Prérequis

Nous vous recommandons vivement d'associer une adresse IP élastique (EIP) à l'instance que vous utilisez pour héberger un WordPress blog. Cela évite à l'adresse DNS publique de votre instance de changer et de détériorer votre installation. Si vous possédez un nom de domaine et que vous voulez l'utiliser pour votre blog, vous pouvez mettre à jour l'enregistrement DNS pour que le nom de domaine pointe vers votre EIP (afin d'obtenir de l'aide à ce sujet, veuillez contacter votre serveur d'inscriptions des noms de domaine). Vous pouvez avoir une EIP associée à une instance en cours d'exécution sans coût aucun. Pour plus d'informations, veuillez consulter la rubrique Adresses IP Elastic dans le Guide de l'utilisateur Amazon EC2. Le didacticiel Tutoriel : Installation d'un serveur LAMP sur AL2023 propose aussi des étapes pour la configuration d'un groupe de sécurité afin d'autoriser le trafic HTTP et HTTPS ainsi que plusieurs étapes afin de vous assurer que les autorisations sur les fichiers sont définies correctement pour votre serveur web. Pour plus d'informations sur l'ajout de règles à votre groupe de sécurité, voir Ajouter des règles à un groupe de sécurité.

Si vous n'avez pas encore de nom de domaine pour votre blog, vous pouvez enregistrer un nom de domaine avec Route 53 et associer l'adresse EIP de votre instance à votre nom de domaine. Pour de plus amples informations, veuillez consulter Inscription de noms de domaines à l'aide d'Amazon Route 53 dans le manuel Amazon Route 53 Manuel du développeur.

Installer WordPress

Connectez-vous à votre instance et téléchargez le package WordPress d'installation. Pour plus d'informations sur la connexion à votre instance, consultez Connexion aux instances AL2023.

1. Téléchargez et installez ces packages à l'aide de la commande suivante.

```
dnf install wget php-mysqlnd httpd php-fpm php-mysqli mariadb105-server php-json
 php php-devel -y
```

2. Vous pouvez remarquer un avertissement affiché avec des commentaires similaires dans la sortie (les versions peuvent varier au fil du temps) :

```
WARNING:
  A newer release of "Amazon Linux" is available.

  Available Versions:

dnf update --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 update --releasever=2023.0.20230204 ... etc
```

   En tant que bonne pratique, nous vous recommandons de conserver le système d'exploitation dans la mesure du up-to-date possible, mais vous souhaiterez peut-être parcourir chaque version pour vous assurer qu'il n'y a aucun conflit dans votre environnement. Si l'installation des packages précédents mentionnés à l'étape 1 échoue, vous devrez peut-être effectuer une mise à jour vers l'une des versions les plus récentes répertoriées, puis réessayer.

3. Téléchargez le dernier package WordPress d'installation à l'aide de la wget commande. La commande suivante devrait toujours télécharger la dernière version.

```
[ec2-user ~]$ wget https://wordpress.org/latest.tar.gz
```

4. Décompressez et désarchivez le package d'installation. Le dossier d'installation est décompressé dans un dossier appelé `wordpress`.

```
[ec2-user ~]$ tar -xzf latest.tar.gz
```

Pour créer un utilisateur de base de données et une base de données pour votre WordPress installation

Votre WordPress installation doit stocker des informations, telles que les articles de blog et les commentaires des utilisateurs, dans une base de données. Cette procédure vous aide à créer la base de données de votre blog et un utilisateur qui est autorisé à lire et à enregistrer des informations dans cette dernière.

1. Démarrez la base de données et le serveur web.

   ```
   [ec2-user ~]$ sudo systemctl start mariadb httpd
   ```

2. Connectez-vous au serveur de base de données en tant qu'utilisateur `root`. Saisissez votre mot de passe `root` de base de données lorsque vous y êtes invité. Il peut être différent du mot de passe `root` de votre système ou il peut même être inexistant si vous n'avez pas sécurisé votre serveur de base de données.

   Si vous n'avez pas encore sécurisé votre serveur de base de données, il est important de le faire. Pour plus d'informations, voir Étape 3 : Sécuriser le serveur de base de données (AL2023).

   ```
   [ec2-user ~]$ mysql -u root -p
   ```

3. Créez un utilisateur et un mot de passe pour votre base de données MySQL. Votre WordPress installation utilise ces valeurs pour communiquer avec votre base de données MySQL. Saisissez la commande suivante en remplaçant les informations par un nom utilisateur et un mot de passe uniques.

   ```
   CREATE USER 'wordpress-user'@'localhost' IDENTIFIED BY 'your_strong_password';
   ```

   Assurez-vous de créer un mot de passe fiable pour votre utilisateur. N'utilisez pas l'apostrophe (') dans votre mot de passe, car elle détériorera la commande précédente. Ne réutilisez pas un mot de passe existant et assurez-vous de stocker ce mot de passe dans un endroit sûr.

4. Créez votre base de données. Donnez à votre base de données un nom descriptif pertinent comme `wordpress-db`.

   > **Note**
   >
   > Les signes de ponctuation autour du nom de la base de données dans la commande ci-dessous sont appelés « accents graves ». La touche « accent grave » (`) est

généralement située au-dessus de la touche Tab d'un clavier QWERTY standard. Les « accents graves » ne sont pas toujours nécessaires, mais ils vous permettent d'utiliser des caractères qui sont normalement interdits dans les noms de base de données, comme les traits d'union.

```
CREATE DATABASE `wordpress-db`;
```

5. Accordez des privilèges complets pour votre base de données à l' WordPress utilisateur que vous avez créé précédemment.

```
GRANT ALL PRIVILEGES ON `wordpress-db`.* TO "wordpress-user"@"localhost";
```

6. Annulez les privilèges de base de données pour récupérer tous vos changements.

```
FLUSH PRIVILEGES;
```

7. Quittez le client `mysql`.

```
exit
```

Pour créer et modifier le fichier wp-config.php

Le dossier WordPress d'installation contient un exemple de fichier de configuration appelé`wp-config-sample.php`. Dans cette procédure, vous copiez ce fichier avant de le modifier pour respecter votre configuration spécifique.

1. Copiez le fichier `wp-config-sample.php` sur un fichier appelé `wp-config.php`. Cela crée un nouveau fichier de configuration et garde le modèle de fichier original intact comme sauvegarde.

```
[ec2-user ~]$ cp wordpress/wp-config-sample.php wordpress/wp-config.php
```

2. Modifiez le fichier `wp-config.php` avec votre éditeur de texte préféré (comme nano ou vim) et saisissez les valeurs pour votre installation. Si vous n'avez pas d'éditeur de texte préféré, `nano` convient aux débutants.

```
[ec2-user ~]$ nano wordpress/wp-config.php
```

a. Trouvez la ligne qui définit DB_NAME et remplacez `database_name_here` par le nom de la base de données que vous avez créée à l'Step 4 de la procédure Pour créer un utilisateur de base de données et une base de données pour votre WordPress installation.

```
define('DB_NAME', 'wordpress-db');
```

b. Trouvez la ligne qui définit DB_USER et remplacez `username_here` par l'utilisateur de base de données que vous avez créé à l'Step 3 de la procédure Pour créer un utilisateur de base de données et une base de données pour votre WordPress installation.

```
define('DB_USER', 'wordpress-user');
```

c. Trouvez la ligne qui définit DB_PASSWORD et remplacez `password_here` par le mot de passe fiable que vous avez créé à l'Step 3 de la procédure Pour créer un utilisateur de base de données et une base de données pour votre WordPress installation.

```
define('DB_PASSWORD', 'your_strong_password');
```

d. Trouvez la section appelée `Authentication Unique Keys and Salts`. Ces SALT valeurs KEY et ces valeurs fournissent une couche de cryptage aux cookies du navigateur que WordPress les utilisateurs stockent sur leurs machines locales. En gros, l'ajout de valeurs longues aléatoires à cet endroit rend votre site plus sécurisé. Consultez https://api.wordpress.org/secret-key/1.1/salt/ pour générer de façon aléatoire un ensemble de valeurs clés que vous pouvez copier et coller dans votre fichier `wp-config.php`. Pour coller du texte dans un terminal PuTTY, placez le curseur où vous voulez coller le texte et faites un clic droit avec votre souris dans le terminal PuTTY.

   Pour plus d'informations sur les clés de sécurité, rendez-vous sur https://wordpress.org/support/article/editing-wp-config-php/#security -keys.

   > **Note**
   >
   > Les valeurs ci-dessous sont proposées à titre d'exemple seulement. N'utilisez pas ces valeurs pour votre installation.

```
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. Enregistrez le fichier et quittez votre éditeur de texte.

Pour installer vos WordPress fichiers sous la racine du document Apache

- Maintenant que vous avez décompressé le dossier d'installation, créé une base de données et un utilisateur MySQL et personnalisé le fichier de WordPress configuration, vous êtes prêt à copier vos fichiers d'installation sur le document root de votre serveur Web afin de pouvoir exécuter le script d'installation qui complète votre installation. L'emplacement de ces fichiers varie selon que vous souhaitez que votre WordPress blog soit disponible à la racine de votre serveur Web (par exemple,*my.public.dns.amazonaws.com*) ou dans un sous-répertoire ou un dossier situé sous la racine (par exemple,*my.public.dns.amazonaws.com/blog*).
  - Si vous souhaitez exécuter WordPress à la racine de votre document, copiez le contenu du répertoire d'installation de WordPress (mais pas le répertoire lui-même) comme suit :

    ```
    [ec2-user ~]$ cp -r wordpress/* /var/www/html/
    ```

  - Si vous WordPress souhaitez exécuter dans un autre répertoire situé sous la racine du document, créez d'abord ce répertoire, puis copiez-y les fichiers. Dans cet exemple, s' WordPressexécutera à partir du répertoire `blog` :

    ```
    [ec2-user ~]$ mkdir /var/www/html/blog
    [ec2-user ~]$ cp -r wordpress/* /var/www/html/blog/
    ```

> **Important**
>
> A des fins de sécurité, si vous ne passez pas à la prochaine procédure immédiatement, arrêtez le serveur web Apache (`httpd`) dès maintenant. Une fois que vous avez déplacé votre installation sous la racine du document Apache, le script WordPress d'installation n'est plus protégé et un attaquant pourrait accéder à votre blog si le serveur Web Apache était en cours d'exécution. Pour arrêter le serveur Web Apache, saisissez la commande sudo service httpd stop. Si vous ne passez pas à la prochaine procédure, vous n'avez pas à arrêter le serveur web Apache.

Pour autoriser l'utilisation WordPress de permaliens

WordPress les permaliens doivent utiliser des `.htaccess` fichiers Apache pour fonctionner correctement, mais cela n'est pas activé par défaut sur Amazon Linux. Utilisez cette procédure pour permettre tous les remplacements à la racine du document Apache.

1. Ouvrez le fichier `httpd.conf` avec votre éditeur de texte préféré (comme nano ou vim). Si vous n'avez pas d'éditeur de texte préféré, `nano` convient aux débutants.

   ```
   [ec2-user ~]$ sudo vim /etc/httpd/conf/httpd.conf
   ```

2. Trouvez la section qui commence par `<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. Modifiez la ligne `AllowOverride None` dans la section ci-dessus par `AllowOverride All`.

> **Note**
>
> Il existe plusieurs lignes `AllowOverride` dans ce fichier. Assurez-vous de modifier la ligne dans la section `<Directory "/var/www/html">`.

```
AllowOverride All
```

4. Enregistrez le fichier et quittez votre éditeur de texte.

Pour installer la bibliothèque de dessins graphiques PHP sur AL2023

La bibliothèque GD pour PHP vous permet de modifier des images. Installez cette bibliothèque si vous devez recadrer l'image d'en-tête pour votre blog. La version phpMyAdmin que vous installez peut nécessiter une version minimale spécifique de cette bibliothèque (par exemple, la version 8.1).

Utilisez la commande suivante pour installer la bibliothèque de dessins graphiques PHP sur AL2023. Par exemple, si vous avez installé php8.1 à partir des sources dans le cadre de l'installation de la pile LAMP, cette commande installe la version 8.1 de la bibliothèque de dessins graphiques PHP.

```
[ec2-user ~]$ sudo dnf install php-gd
```

Pour vérifier la version installée, utilisez la commande suivante :

```
[ec2-user ~]$ sudo dnf list installed | grep php-gd
```

Voici un exemple de sortie :

```
php-gd.x86_64                    8.1.30-1.amzn2           @amazonlinux
```

Pour installer la bibliothèque de dessins graphiques PHP sur Amazon Linux AMI

La bibliothèque GD pour PHP vous permet de modifier des images. Installez cette bibliothèque si vous devez recadrer l'image d'en-tête pour votre blog. La version phpMyAdmin que vous installez peut nécessiter une version minimale spécifique de cette bibliothèque (par exemple, la version 8.1).

Pour vérifier quelles versions sont disponibles, utilisez la commande suivante :

```
[ec2-user ~]$ dnf list | grep php
```

Voici un exemple de lignes de sortie de la bibliothèque de dessin graphique PHP (version 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
```

Utilisez la commande suivante pour installer une version spécifique de la bibliothèque de dessin graphique PHP (par exemple, la version php8.1) sur l'AMI Amazon Linux :

```
[ec2-user ~]$ sudo dnf install -y php8.1-gd
```

Pour corriger les autorisations sur les fichiers pour le serveur web Apache

Certaines des fonctionnalités disponibles WordPress nécessitent un accès en écriture à la racine du document Apache (comme le téléchargement de médias via les écrans d'administration). Si vous ne l'avez pas déjà fait, appliquez les autorisations et appartenances aux groupes suivantes (comme décrit plus en détail dans le tutoriel sur le serveur web LAMP).

1. Accordez la propriété du fichier `/var/www` et de son contenu à l'utilisateur `apache`.

```
[ec2-user ~]$ sudo chown -R apache /var/www
```

2. Accordez la propriété de groupe de `/var/www` et de son contenu au groupe `apache`.

```
[ec2-user ~]$ sudo chgrp -R apache /var/www
```

3. Modifiez les autorisations sur les répertoires de `/var/www` et ses sous-répertoires pour ajouter des autorisations d'écriture de groupe et définir l'ID de groupe pour les futurs sous-répertoires.

```
[ec2-user ~]$ sudo chmod 2775 /var/www
[ec2-user ~]$ find /var/www -type d -exec sudo chmod 2775 {} \;
```

4. Modifiez de façon récursive les autorisations sur les fichiers de `/var/www` et ses sous-répertoires.

```
[ec2-user ~]$ find /var/www -type f -exec sudo chmod 0644 {} \;
```

> **Note**
>
> Si vous avez l'intention de l'utiliser également WordPress en tant que serveur FTP, vous aurez besoin de paramètres de groupe plus permissifs ici. Pour ce faire, veuillez consulter les étapes recommandées et WordPress les paramètres de sécurité.

5. Redémarrez le serveur web Apache pour récupérer les nouveaux groupe et autorisations.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

Pour exécuter le script WordPress d'installation avec AL2023

Vous êtes prêt à procéder à l'installation WordPress. Les commandes que vous utilisez dépendent du système d'exploitation. Les commandes de cette procédure sont destinées à être utilisées avec AL2023. Utilisez la procédure qui suit celle-ci avec l'AMI AL2023.

1. Utilisez la commande systemctl pour vous assurer que les services `httpd` et de base de données commencent à chaque démarrage système.

```
[ec2-user ~]$ sudo systemctl enable httpd && sudo systemctl enable mariadb
```

2. Vérifiez que le serveur de base de données est en cours d'exécution.

```
[ec2-user ~]$ sudo systemctl status mariadb
```

   Si le service de base de données n'est pas en cours d'exécution, démarrez-le.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

3. Vérifiez que votre serveur web Apache (`httpd`) est en cours d'exécution.

```
[ec2-user ~]$ sudo systemctl status httpd
```

   Si le service `httpd` n'est pas en cours d'exécution, démarrez-le.

```
[ec2-user ~]$ sudo systemctl start httpd
```

4. Dans un navigateur Web, saisissez l'URL de votre WordPress blog (soit l'adresse DNS publique de votre instance, soit cette adresse suivie du `blog` dossier). Vous devriez voir le script WordPress d'installation. Fournissez les informations requises par l' WordPress installation. Choisissez Installer WordPress pour terminer l'installation. Pour plus d'informations, voir Étape 5 : Exécuter le script d'installation sur le WordPress site Web.

Pour exécuter le script WordPress d'installation avec l'AMI AL2023

1. Utilisez la commande chkconfig pour vous assurer que les services `httpd` et de base de données commencent à chaque démarrage système.

```
[ec2-user ~]$ sudo chkconfig httpd on && sudo chkconfig mariadb on
```

2. Vérifiez que le serveur de base de données est en cours d'exécution.

```
[ec2-user ~]$ sudo service mariadb status
```

   Si le service de base de données n'est pas en cours d'exécution, démarrez-le.

```
[ec2-user ~]$ sudo service mariadb start
```

3. Vérifiez que votre serveur web Apache (`httpd`) est en cours d'exécution.

```
[ec2-user ~]$ sudo service httpd status
```

Si le service `httpd` n'est pas en cours d'exécution, démarrez-le.

```
[ec2-user ~]$ sudo service httpd start
```

4. Dans un navigateur Web, saisissez l'URL de votre WordPress blog (soit l'adresse DNS publique de votre instance, soit cette adresse suivie du `blog` dossier). Vous devriez voir le script WordPress d'installation. Fournissez les informations requises par l' WordPress installation. Choisissez Installer WordPress pour terminer l'installation. Pour plus d'informations, voir Étape 5 : Exécuter le script d'installation sur le WordPress site Web.

Étapes suivantes

Après avoir testé votre WordPress blog, pensez à mettre à jour sa configuration.

Utiliser un nom de domaine personnalisé

Si vous avez un nom de domaine associé à l'EIP de votre instance EC2, vous pouvez configurer votre blog pour utiliser ce nom au lieu de l'adresse DNS publique EC2. Pour plus d'informations, voir Modification de l'URL du site sur le WordPress site Web.

Configurer votre blog

Vous pouvez configurer votre blog pour utiliser différents thèmes et plugins afin de proposer une expérience plus personnalisée à vos lecteurs. Cependant, il peut arriver que le processus d'installation échoue ce qui entraînera la perte de tout votre blog. Nous vous recommandons vivement de créer une sauvegarde de l'Amazon Machine Image (AMI) de votre instance avant d'essayer d'installer des thèmes ou des plugins. Ainsi, vous pouvez restaurer votre blog en cas de problème pendant l'installation. Pour plus d'informations, consultez la section Création de votre propre AMI dans le guide de l'utilisateur Amazon EC2.

Augmenter la capacité

Si votre WordPress blog devient populaire et que vous avez besoin de plus de puissance de calcul ou de stockage, considérez les étapes suivantes :

- Développez l'espace de stockage sur votre instance. Pour plus d'informations, consultez Amazon EBS Elastic Volumes.

- Déplacez votre base de données MySQL vers Amazon RDS pour profiter de la capacité du service à se mettre à l'échelle facilement.

Améliorer les performances réseau de votre trafic Internet

Si vous vous attendez à ce que votre blog génère du trafic provenant d'utilisateurs situés dans le monde entier, envisagez d'utiliser AWS Global Accelerator. Global Accelerator vous aide à réduire le temps de latence en améliorant les performances du trafic Internet entre les appareils clients de vos utilisateurs et votre WordPress application en cours d'exécution AWS. Global Accelerator utilise le réseau AWS mondial pour diriger le trafic vers un point de terminaison d'application sain dans la AWS région la plus proche du client.

En savoir plus sur WordPress

Les liens suivants contiennent plus d'informations sur WordPress.

- Pour plus d'informations WordPress, consultez la documentation d'aide du WordPress Codex sur le site du Codex.
- Pour plus d'informations sur le dépannage de votre installation, consultez la section Problèmes d'installation courants.
- Pour plus d'informations sur la manière de sécuriser votre WordPress blog, consultez la section Renforcement. WordPress
- Pour plus d'informations sur la gestion de votre WordPress blog up-to-date, consultez la section Mise à jour WordPress.

Aide! Mon nom DNS public a changé et mon blog ne fonctionne plus

Votre WordPress installation est automatiquement configurée à l'aide de l'adresse DNS publique de votre instance EC2. Si vous arrêtez et redémarrez l'instance, l'adresse DNS publique change (à moins qu'elle soit associée à une adresse IP Elastic) et votre blog ne fonctionne plus, car il fait référence à des ressources à une adresse qui n'existe plus (ou qui est assignée à une autre instance EC2). Une description plus détaillée du problème et plusieurs solutions possibles sont présentées dans https://wordpress.org/support/article/changing-the-site-url/.

Si cela est arrivé à votre WordPress installation, vous pourrez peut-être récupérer votre blog en suivant la procédure ci-dessous, qui utilise l'interface de ligne de wp-cli commande pour WordPress.

Pour modifier l'URL de votre WordPress site à l'aide du wp-cli

1. Connectez-vous à votre instance EC2 avec SSH.
2. Notez l'ancienne URL de site et la nouvelle URL de site pour votre instance. L'ancienne URL du site est probablement le nom DNS public de votre instance EC2 lors de l'installation WordPress. La nouvelle URL de site correspond au nom DNS public actuel pour votre instance EC2. Si vous n'êtes pas certain de votre ancienne URL de site, vous pouvez utiliser curl pour la trouver avec la commande suivante.

   ```
   [ec2-user ~]$ curl localhost | grep wp-content
   ```

   Vous devriez voir des références à votre ancien nom DNS public dans les données de sortie qui ressembleront à cela (ancienne URL de site en rouge) :

   ```
   <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. Téléchargez le kit wp-cli avec la commande suivante.

   ```
   [ec2-user ~]$ curl -O https://raw.githubusercontent.com/wp-cli/builds/gh-pages/
   phar/wp-cli.phar
   ```
4. Recherchez et remplacez l'ancienne URL du site dans votre WordPress installation à l'aide de la commande suivante. Remplacez l'ancienne et la nouvelle URL du site par votre instance EC2 et le chemin d'accès à votre WordPress installation (généralement `/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. Dans un navigateur Web, entrez l'URL du nouveau site de votre WordPress blog pour vérifier que le site fonctionne à nouveau correctement. Si ce n'est pas le cas, consultez les sections Modification de l'URL du site et Problèmes d'installation courants pour plus d'informations.

Utilisation d'Amazon Linux 2023 en dehors d'Amazon EC2

Les images de conteneurs Amazon Linux 2023 peuvent être exécutées dans des environnements d'exécution de conteneurs compatibles. Pour plus d'informations sur l'utilisation d'Amazon Linux 2023 dans un conteneur, consultez AL2023 dans des conteneurs.

Amazon Linux 2023 (AL2023) peut également être exécuté en tant qu'invité virtualisé en dehors de l'exécution directe sur Amazon EC2. Des images KVM (`qcow2`), VMware (`OVA`) et Hyper-V (`vhdx`) sont actuellement disponibles.

> **Note**
>
> La configuration des images Amazon Linux 2023 est différente de celle d'Amazon Linux 2. Si vous venez d'Exécution d'Amazon Linux 2 en tant que machine virtuelle sur site, vous devrez adapter votre configuration pour qu'elle soit compatible avec AL2023.

Téléchargez des images Amazon Linux 2023 à utiliser avec KVM, VMware et Hyper-V

Les images de disque Amazon Linux 2023 destinées à être utilisées avec KVM, VMware et Hyper-V peuvent être téléchargées sur cdn.amazonlinux.com.

Configurations prises en charge d'Amazon Linux 2023 pour une utilisation dans des environnements virtualisés autres qu'Amazon EC2

Cette section décrit les exigences relatives à l'exécution d'Amazon Linux 2023 dans des environnements virtualisés autres qu'Amazon EC2, tels que KVM, VMware ou Hyper-V.

La Configuration système requise pour AL2023 de base s'applique à tous les environnements virtualisés autres qu'Amazon EC2. La liste des modèles d'appareils pris en charge est détaillée pour chaque environnement d'hyperviseur dans les rubriques suivantes.

KVM, VMware et Hyper-V proposent de nombreuses options de configuration, et il convient de les configurer avec soin afin de répondre à vos besoins en matière de sécurité, de performances et de fiabilité. Pour plus d'informations, consultez la documentation fournie par votre hyperviseur.

Rubriques

Exigences pour exécuter AL2023 sur KVM

Cette section décrit les conditions requises pour exécuter AL2023 sur KVM. Les images KVM d'AL2023 sont disponibles pour les architectures `aarch64` et `x86-64`. Ces exigences s'ajoutent à la base Configuration système requise pour AL2023 pour les images KVM.

Rubriques

Exigences relatives à l'hôte KVM pour exécuter AL2023 sur KVM

Les images KVM sont actuellement qualifiées sur un hôte exécutant Ubuntu 22.04.3 LTS avec la `qemu` version6.2+dfsg-2ubuntu6.15, fournie par cette version d'Ubuntu, utilisant un type de `q35` machine pour `x86-64` et un type de `virt` machine pour. `aarch64`

Support des appareils pour AL2023 sur KVM

Les modèles d'appareils **`qemu`** testés pour une utilisation avec les images KVM AL2023 (**`aarch64`** et **`x86-64`**) sont les suivants :

- `virtio-blk` (périphérique de stockage en mode bloc `virtio`)
- `virtio-scsi` (contrôleur `virtio` SCSI avec périphérique à disque)
- `virtio-net` (périphérique réseau `virtio`)
- `ahci` (à utiliser avec le lecteur de CD-ROM virtuel)

- `usb-storage` (via `xhci`)

Les modèles **`qemu`** d'appareils supplémentaires compatibles avec la qualification d'image KVM AL2023, mais peu sollicités, sont les suivants :

- `VGA` (`qemu` VGA) sur `x86-64` uniquement
- `virtio-rng` (générateur virtuel de nombres aléatoires)
- appareils à clavier AT et souris PS/2 hérités
- appareil série hérité

Mode de démarrage (UEFIetBIOS) support pour AL2023 sur KVM

L'image `x86-64` est testée en modes de démarrage hérités BIOS et UEFI. Les images `aarch64` sont testées en mode démarrage UEFI.

> **Note**
>
> Par défaut, lors de l'utilisation du mode de UEFI démarrage, certains gestionnaires de machines virtuelles fournissent à la machine virtuelle des clés Microsoft Secure Boot qui activent le démarrage sécurisé. Cette configuration ne démarre pas AL2023.
> Le chargeur de démarrage AL2023 n'étant pas signé par Microsoft, la machine virtuelle doit être provisionnée soit sans clés UEFI, soit avec les clés AL2023 pour le démarrage sécurisé.

> **Important**
>
> La prise en charge du démarrage sécurisé pour les KVM images n'a pas encore été validée.

Limitations relatives à l'exécution d'AL2023 sur KVM

L'exécution d'AL2023 sur KVM présente certaines limites connues.

> **Note**
>
> Le code implémentant certaines des fonctionnalités non prises en charge répertoriées peut exister dans AL2023 et fonctionner correctement. La liste des fonctionnalités non prises en

charge existe afin que vous puissiez prendre des décisions éclairées sur les fonctionnalités sur lesquelles vous pouvez compter aujourd'hui et sur celles que l'équipe Amazon Linux qualifiera de efficaces dans le cadre des futures mises à jour.

Limitations connues relatives à l'exécution d'AL2023 sur KVM

- L'agent invité KVM n'est actuellement ni proposé en package ni pris en charge.
- Le branchement et le débranchement à chaud du processeur, de la mémoire ou de tout autre type d'appareil ne sont pas pris en charge.
- L'hibernation des machines virtuelles n'est pas prise en charge.
- La migration des machines virtuelles n'est pas prise en charge.
- La transmission d'un appareil via PCI Passthrough ou USB Passthrough n'est pas prise en charge.

Conditions requises pour exécuter AL2023 sur VMware

Cette section décrit les conditions requises pour exécuter AL2023 surVMware. Les VMware images d'AL2023 ne sont disponibles que pour l'`x86-64`architecture. VMwareles images pour ne `aarch64` sont pas disponibles ou prises en charge. Ces exigences s'ajoutent à la base Configuration système requise pour AL2023 des VMware images.

Rubriques

- VMwareexigences relatives à l'hôte pour exécuter AL2023 sur VMware
- Support de l'appareil pour AL2023 sur VMware
- Mode de démarrage (UEFIetBIOS) support pour AL2023 sur VMware
- Limitations relatives à l'exécution d'AL2023 sur VMware

VMwareexigences relatives à l'hôte pour exécuter AL2023 sur VMware

Les images VMware OVA de l'AL2023 sont actuellement qualifiées selon les critères suivants :

- VMwareStation de travail 17.5.0 exécutée sur des hôtes utilisant un processeur Intel (R) Xeon (R) Platinum 8124M
- VMwarevSphere 8.0 utilisant un processeur Intel (R) Xeon (R) Platinum 8275CL

Les images VMware OVA AL2023 indiquent une version matérielle de la machine de 13.

VMwareLa version 13 du matériel de la machine est prise en charge par :

- ESXi 6.5 ou version ultérieure
- VMwareWorkstation 14 ou version ultérieure

Support de l'appareil pour AL2023 sur VMware

Les modèles d'VMwareappareils suivants ont été testés pour une utilisation avec des images AL2023 VMware OVA (**`x86-64`**uniquement) :

- `vmw_pvscsi`(contrôleur VMware paravirtualiséSCSI)
- `vmxnet3`(périphérique VMware réseau paravirtualisé)
- `ata_piix` (IDE hérité à utiliser avec le lecteur de CD-ROM virtuel)

Modèles d'VMwareappareils supplémentaires activés dans le cadre de la qualification VMware d'image AL2023, mais peu sollicités :

- `vmw_vmci`et `vsock` interface associée (transport de socket virtuel pour l'agent VMware invité)
- Dispositif à ballon mémoire `vmw_balloon`
- VMware`SVGA`contrôleur
- appareils à clavier AT et souris PS/2 hérités

Le package VMware d'agent invité (`open-vm-tools`) est disponible et installé par défaut dans les images VMware OVA AL2023.

Mode de démarrage (UEFIetBIOS) support pour AL2023 sur VMware

Depuis la version 2023.3.20231211, l'image VMware OVA AL2023 a été validée en mode ancien et en mode démarrage. BIOS UEFI La configuration par défaut OVA est toujours existante BIOS mais peut être modifiée par l'utilisateur.

> **Important**
>
> La prise en charge du démarrage sécurisé nécessiteUEFI, ce qui n'a pas été validé pour l'exécution d'AL2023 surVMware.

Limitations relatives à l'exécution d'AL2023 sur VMware

Il existe certaines limites connues liées à l'exécution d'AL2023 surVMware.

> **Note**
>
> Le code implémentant certaines des fonctionnalités non prises en charge répertoriées peut exister dans AL2023 et fonctionner correctement. La liste des fonctionnalités non prises en charge existe afin que les clients puissent prendre des décisions éclairées sur les fonctionnalités sur lesquelles ils peuvent compter aujourd'hui et sur celles que l'équipe Amazon Linux décrit comme fonctionnant dans le cadre des futures mises à jour.

Limitations connues liées à l'exécution d'AL2023 sur VMware

- UEFISecure Boot n'est actuellement pas validé avec AL2023 activéVMware.
- Le branchement et le débranchement à chaud du processeur, de la mémoire ou de tout autre type d'appareil ne sont pas pris en charge.
- L'hibernation des machines virtuelles n'est pas prise en charge.
- La migration des machines virtuelles n'est pas prise en charge.
- La transmission d'un appareil via PCI Passthrough ou USB Passthrough n'est pas prise en charge.

Conditions requises pour exécuter Amazon Linux 2023 sur Hyper-V

Cette section décrit les conditions requises pour exécuter Amazon Linux 2023 sur Hyper-V. Les images Hyper-V d'AL2023 ne sont disponibles que pour l'`x86-64`architecture. Les images Hyper-V pour ne `aarch64` sont pas disponibles ou prises en charge pour le moment.

Cette section couvre les exigences supplémentaires en plus de la base Configuration système requise pour AL2023 pour les images Hyper-V.

Rubriques

- Exigences relatives à l'hôte Hyper-V pour exécuter Amazon Linux 2023 sur Hyper-V
- Support des appareils pour Amazon Linux 2023 sur Hyper-V
- Limitations relatives à l'exécution d'Amazon Linux 2023 sur Hyper-V

Exigences relatives à l'hôte Hyper-V pour exécuter Amazon Linux 2023 sur Hyper-V

La principale qualification d'Amazon Linux 2023 sur Hyper-V se produit sur Windows Server 2022 exécuté sur une instance EC2`c5.metal`.

Support des appareils pour Amazon Linux 2023 sur Hyper-V

Amazon Linux 2023 est testé sur des machines virtuelles Hyper-V de génération 1 et de génération 2 avec l'ensemble de matériel virtualisé suivant :

- Machine virtuelle de génération 1 (démarrage du BIOS traditionnel)
- Machine virtuelle de génération 2 (démarrage UEFI, pas de démarrage sécurisé)
- Les modèles d'appareils suivants ont été testés pour être utilisés avec les images Hyper-V AL2023 :
  - Stockage virtuel Hyper-V *`hv_storvsc`* pour le disque racine et le lecteur de CD-ROM émulé sur les machines virtuelles de génération 2
  - IDE PIIX émulé *`ata_piix`* pour le lecteur de CD-ROM virtuel sur les machines virtuelles de génération 1
  - Ethernet virtuel Hyper-V `hv_netvsc`
- Les modèles d'appareils suivants sont activés mais légèrement testés :
  - Mode texte VGA classique sur les machines virtuelles de génération 1
  - Framebuffer basé sur le microprogramme UEFI sur les machines virtuelles de *`simpledrmfb`* génération 2
  - Ballon Hyper-V `hv_balloon`
  - Ballon Hyper-V `hv_balloon`
  - Hyper-V HID/souris `hid_hyperv`
- Les modes de périphérique suivants ne sont pas activés dans AL2023 pour le moment :
  - Transfert PCI Hyper-V
  - Carte graphique Hyper-V DRM

> **Important**
>
> Pour les machines virtuelles de génération 2, le démarrage sécurisé n'est pas pris en charge et doit être désactivé avant le lancement de la machine virtuelle pour un démarrage réussi d'Amazon Linux 2023. Hyper-V ne prend actuellement en charge le démarrage sécurisé

qu'avec des composants logiciels signés par les propres clés de Microsoft, tandis que le chargeur de démarrage Amazon Linux est signé par une clé privée Amazon. Hyper-V ne prend pas en charge l'importation de clés tierces pour le moment.

Limitations relatives à l'exécution d'Amazon Linux 2023 sur Hyper-V

Voici quelques limitations connues liées à l'exécution d'Amazon Linux 2023 sur Hyper-V :

> **Note**
>
> Le code implémentant certaines des fonctionnalités non prises en charge répertoriées peut exister dans AL2023 et fonctionner correctement. La liste des fonctionnalités non prises en charge existe afin que les clients puissent prendre des décisions éclairées sur les fonctionnalités sur lesquelles ils peuvent compter aujourd'hui et sur celles que l'équipe Amazon Linux décrit comme fonctionnant dans le cadre des futures mises à jour.

Limitations connues liées à l'exécution d'AL2023 sur Hyper-V

- Le mode de démarrage sécurisé UEFI n'est actuellement pas pris en charge ni fonctionnel avec AL2023 sur Hyper-V
- Le branchement et le débranchement à chaud du processeur, de la mémoire ou de tout autre type d'appareil ne sont pas pris en charge.
- L'hibernation de machine virtuelle (VM) n'est pas prise en charge.
- La migration de machine virtuelle (VM) n'est pas prise en charge.
- La transmission d'un appareil via PCI Passthrough ou USB Passthrough n'est pas prise en charge.

Installation et configuration **`cloud-init`** d'Amazon Linux 2023 en cas d'utilisation en dehors d'Amazon EC2

Cette section explique comment installer et configurer une machine virtuelle Amazon Linux 2023 lorsqu'elle n'est pas exécutée directement sur Amazon EC2, par exemple sur KVM, VMware ou Hyper-V.

Par défaut, les images d'une machine virtuelle Amazon Linux 2023 ne sont fournies avec aucun mot de passe utilisateur ni clé ssh et obtiennent leur configuration réseau via DHCP sur la première

interface réseau découverte. Cela signifie que par défaut, sans configuration supplémentaire, il n'est pas possible de se connecter à la machine virtuelle résultante.

Ainsi, une certaine forme de configuration doit être fournie à la machine virtuelle. Le mécanisme standard pour effectuer cette opération pour Amazon Linux consiste à utiliser des sources de données `cloud-init`.

Amazon Linux 2023 a été qualifié avec les sources de données suivantes :

NoCloud

Il s'agit de la méthode traditionnelle de configuration d'images sur site via un CD-ROM virtuel contenant une image ISO9660 de départ avec des fichiers de configuration `cloud-init`.

VMware

Amazon Linux 2023 prend également en charge la configuration d'images VMware exécutées sur vSphere via la source de données spécifique à VMware via VMware `guestinfo.userdata` et `guestinfo.metadata`.

> **Note**
>
> La configuration des sources de données peut être différente de celle d'Amazon Linux 2. Plus précisément, Amazon Linux 2023 utilise `systemd-networkd` pour sa configuration et nécessite l'utilisation de « Networking Config Version 2 » de `cloud-init`, comme indiqué dans la documentation sur la configuration réseau de `cloud-init`.

La documentation complète des mécanismes de configuration de `cloud-init` pour la version de `cloud-init` fournie en package dans Amazon Linux 2023 se trouve dans la documentation en amont pour `cloud-init`.

NoCloud (`seed.iso`) `cloud-init` configuration pour Amazon Linux 2023 sur KVM et VMware

Cette section explique comment créer et utiliser une `seed.iso` image pour configurer Amazon Linux 2023 exécuté sur KVM ouVMware. Étant donné que KVM les VMware environnements ne disposent pas du service de métadonnées d'instance Amazon EC2 (IMDS), une autre méthode de configuration d'Amazon Linux 2023 est requise, et la fourniture d'une `seed.iso` image est l'une de ces méthodes.

L'image de démarrage `seed.iso` inclut les informations de configuration initiale requises pour démarrer votre nouvelle machine virtuelle, telles que la configuration réseau, le nom d'hôte et les données utilisateur.

> **Note**
>
> L'image `seed.iso` inclut uniquement les informations de configuration requises pour démarrer la machine virtuelle. Elle n'inclut pas les fichiers de système d'exploitation Amazon Linux 2023.

Pour générer l'image `seed.iso`, vous avez besoin de deux fichiers de configuration, parfois trois :

`meta-data`

Ce fichier inclut généralement le nom d'hôte de la machine virtuelle.

`user-data`

Ce fichier configure les comptes d'utilisateur et spécifie leurs mots de passe, paires de clés ssh et/ou mécanismes d'accès. Par défaut, les images Amazon Linux 2023 KVM et VMware créent un compte d'utilisateur `ec2-user`. Vous utilisez le fichier de configuration `user-data` pour définir le mot de passe et/ou les clés ssh pour ce compte d'utilisateur par défaut.

`network-config` (facultatif)

Ce fichier fournit généralement une configuration réseau pour la machine virtuelle qui remplacera celle par défaut. La configuration par défaut consiste à utiliser DHCP sur la première interface réseau disponible.

Création de l'image de disque **`seed.iso`**

1. Sur Linux ou macOS, créez un nouveau dossier appelé `seedconfig` et accédez à celui-ci.

   > **Note**
   >
   > Il est possible d'utiliser Windows ou un autre système d'exploitation pour effectuer ces étapes, mais vous devrez rechercher l'outil équivalent à `mkisofs` pour terminer la création de l'image `seed.iso`.

2. Créez le fichier de configuration `meta-data`.

   a. Créez un nouveau fichier nommé `meta-data`.

   b. Ouvrez le fichier `meta-data` à l'aide de l'éditeur de votre choix et ajoutez ce qui suit, en remplaçant *`vm-hostname`* par le nom d'hôte de la machine virtuelle :

   ```
   local-hostname: vm-hostname
   ```

   c. Enregistrez et fermez le fichier de configuration `meta-data`.

3. Créez le fichier de configuration `user-data`.

   a. Créez un nouveau fichier nommé `user-data`.

   b. Ouvrez le fichier `user-data` à l'aide de l'éditeur de votre choix et ajoutez ce qui suit, en effectuant les remplacements nécessaires :

   ```
   #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. Vous pouvez éventuellement ajouter d'autres comptes utilisateurs au fichier `user-data` de configuration.

   De même, vous pouvez créer des comptes d'utilisateur supplémentaires et spécifier leurs mécanismes d'accès, mots de passe et paires de clés. Pour plus d'informations sur les directives prises en charge, consultez la documentation en amont pour `cloud-init`.

   d. Enregistrez et fermez le fichier de configuration `user-data`.

4. (Facultatif) Créez le fichier de configuration `network-config`.

   a. Créez un nouveau fichier nommé `network-config`.

   b. Ouvrez le fichier `network-config` à l'aide de l'éditeur de votre choix et ajoutez ce qui suit, en remplaçant les différentes adresses IP par celles qui conviennent à votre configuration.

```
version: 2
ethernets:
  enp1s0:
    addresses:
      - 192.168.122.161/24
    gateway4: 192.168.122.1
    nameservers:
      addresses: 192.168.122.1
```

> **Note**
>
> La configuration réseau `cloud-init` fournit des mécanismes permettant de faire correspondre l'adresse de l'interface MAC au lieu de spécifier le nom de l'interface, qui peut changer en fonction de la configuration de la machine virtuelle. Ces fonctionnalités `cloud-init` (et bien d'autres) de configuration réseau sont décrites plus en détail dans la documentation Network Config Version 2 en amont pour `cloud-init`.

c. Enregistrez et fermez le fichier de configuration `network-config`.

5. Créez l'image disque `seed.iso` à l'aide des fichiers de configuration `meta-data`, `user-data` et facultatifs `network-config` lors des étapes précédentes.

   Selon le système d'exploitation sur lequel vous créez l'image de disque `seed.iso`, procédez comme indiqué ci-après.

   - Sur les systèmes Linux, utilisez un outil comme **`mkisofs`** ou **`genisoimage`** pour créer le fichier complet `seed.iso`. Accédez au dossier `seedconfig` et exécutez la commande suivante :

     ```
     $ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data
     ```

   - Si vous utilisez un `network-config`, incluez-le dans l'invocation de **`mkisofs`** :

     ```
     $ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data
      network-config
     ```

   - Sur les systèmes macOS, vous pouvez utiliser un outil comme **`hdiutil`** pour générer le fichier final `seed.iso`. Comme **`hdiutil`** utilise un nom de chemin plutôt qu'une liste de

fichiers, la même invocation peut être utilisée, qu'un fichier de configuration `network-config` ait été créé ou non.

```
$ hdiutil makehybrid -o seed.iso -hfs -joliet -iso -default-volume-name cidata
 seedconfig/
```

6. Le fichier `seed.iso` obtenu peut désormais être joint à la nouvelle machine virtuelle Amazon Linux 2023 via un lecteur de CD-ROM virtuel afin que `cloud-init` le trouve au premier démarrage et applique la configuration au système.

VMware**`cloud-init`**configuration guestinfo pour AL2023 sur VMware

VMwareles environnements ne disposent pas du service de métadonnées d'instance Amazon EC2 (IMDS), une autre méthode de configuration de l'AL2023 est donc requise. Cette section décrit comment utiliser un mécanisme de configuration alternatif au lecteur de CD-ROM `seed.iso` virtuel disponible dans VMware vSphere.

Cette méthode de configuration utilise le VMware `extraconfig` mécanisme pour fournir des données de configuration à`cloud-init`. Pour chacune des clés suivantes, une ***`keyname`*`.encoding`** propriété correspondante doit être fournie.

Les clés suivantes peuvent être fournies au VMware `extraconfig` mécanisme.

`guestinfo.metadata`

JSON ou YAML contenant des métadonnées `cloud-init`

`guestinfo.userdata`

Un document YAML contenant des données utilisateur `cloud-init` au format `cloud-config`.

`guestinfo.vendordata` (facultatif)

YAMLcontenant les données du `cloud-init` fournisseur

Les propriétés d'encodage correspondantes (`guestinfo.metadata.encoding`, `guestinfo.userdata.encoding` et `guestinfo.vendordata.encoding`) peuvent contenir :

`base64`

Le contenu de la propriété est encodé en `base64`.

`gzip+base64`

Le contenu de la propriété est compressé avec `gzip` après l'encodage en `base64`.

> **Note**
>
> La `seed.iso` méthode prend en charge un fichier de `network-config` configuration distinct (facultatif). VMware`guestinfo`diffère dans la manière dont la configuration réseau est fournie. Des informations supplémentaires sont fournies dans la section suivante.

Si une configuration réseau explicite est souhaitée, elle doit être intégrée dans `metadata` sous la forme de deux propriétés YAML ou JSON :

`network`

Contient la configuration réseau codée au format JSON ou YAML.

`network.encoding`

Contient le codage des données de configuration réseau ci-dessus. Les encodages `cloud-init` pris en charge sont les mêmes que pour les données `guestinfo` : `base64` et `gzip+base64`.

Example Utilisation de l'outil VMware vSphere **`govc`** CLI pour transmettre la configuration avec **`guestinfo`**

1. Préparez les fichiers de `network-config` configuration `meta-data``user-data`, et facultatifs, comme décrit dansNoCloud (`seed.iso`) `cloud-init` configuration pour Amazon Linux 2023 sur KVM et VMware.
2. Convertissez les fichiers de configuration dans des formats utilisables par 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"
```

Comparaison des packages installés sur l'AMI standard Amazon Linux 2023 avec l'image KVM AL2023

Comparaison des RPM présents sur l'AMI standard AL2023 par rapport aux RPM présents sur l'image KVM AL2023.

Package	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	2.4.1	
amazon-linux-onprem		1.2
amazon-linux-repo-cdn		2023,4.20240513
amazon-linux-repo-s3	2023,4.20240513	

Package	AMI	KVM
amazon-linux-sb-keys	2023.1	2023.1
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3,3.380,0	3,3.380,0
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	2,15,30	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	9,16,48	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

Package	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	2023,2,64	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
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	20220428	20220428

Package	AMI	KVM
crypto-policies-scripts	20220428	20220428
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	2,127	2,127
cyrus-sasl-plain	2,127	2,127
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

Package	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

Package	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

Package	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 (aarch64)	2,06 (aarch64)
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 (novembre)
grub2-tools	2,06	2,06

Package	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	0,353	0,353
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	

Package	AMI	KVM
json-c	0,14	0,14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6,1,90	6,1,90
kernel-livepatch-repo-cdn		2023,4.20240513
kernel-livepatch-repo-s3	2023,4.20240513	
kernel-modules-extra		6,1,90
kernel-modules-extra-common		6,1,90
kernel-srpm-macros	1.0	1.0
kernel-tools	6,1,90	6,1,90
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

Package	AMI	KVM
libaio	0,3,111	0,3,111
libarchive	3.5.3	3.5.3
libargon2	27/12/2017	27/12/2017
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0,11	0,11
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
libeconf	0,4,0	0,4,0
libedit	3.1	3.1

Package	AMI	KVM
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	0,13	0,13
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2,37,4	2,37,4

Package	AMI	KVM
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	0,15	0,15
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

Package	AMI	KVM
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
libverto	0,3.2	0,3.2
libverto-libev	0,3.2	0,3.2

Package	AMI	KVM
libxcrypt	4.4.33	4.4.33
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
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
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8

Package	AMI	KVM
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	8,7 p1	8,7 p1
openssh-clients	8,7 p1	8,7 p1
openssh-server	8,7 p1	8,7 p1
openssl	3,0.8	3,0.8

Package	AMI	KVM
openssl-libs	3,0.8	3,0.8
openssl-pkcs11	0,4,12	0,4,12
os-prober	1,77	1,77
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	10,40	10,40
pcre2-syntax	10,40	10,40
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

Package	AMI	KVM
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
perl-parent	0,238	0,238
perl-PathTools	3,78	3,78
perl-Pod-Escapes	1,07	1,07

Package	AMI	KVM
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	2021,0726	2021,0726
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

Package	AMI	KVM
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	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

Package	AMI	KVM
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
python3-libs	3,9,16	3,9,16
python3-libselinux	3.4	3.4
python3-libsemanage	3.4	3.4

Package	AMI	KVM
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0,1,6	0,1,6
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
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

Package	AMI	KVM
python3-ruamel-yaml	0,16.6	0,16.6
python3-ruamel-yaml-clib	0,12	0,12
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
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

Package	AMI	KVM
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	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	

Package	AMI	KVM
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	252,16	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
systemd-udev	252,16	252,16
system-release	2023,4.20240513	2023,4.20240513
systemtap-runtime	4.8	4.8
tar	1,34	1,34
tbb	2020,3	2020,3
tcpdump	4,99,1	4,99,1
tcsh	6,24,07	6,24,07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0

Package	AMI	KVM
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
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0

Package	AMI	KVM
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

Comparaison des packages installés sur l'AMI standard Amazon Linux 2023 avec l'image VMWare OVA AL2023

Comparaison des RPM présents sur l'AMI standard AL2023 par rapport aux RPM présents sur l'image VMware OVA AL2023.

Package	AMI	VMware OVA
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		2023,4.20240513
amazon-linux-repo-s3	2023,4.20240513	
amazon-linux-sb-keys	2023.1	2023.1

Package	AMI	VMware OVA
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3,3.380,0	3,3.380,0
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	2,15,30	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	9,16,48	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

Package	AMI	VMware OVA
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
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	20220428	20220428
crypto-policies-scripts	20220428	20220428

Package	AMI	VMware OVA
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	2,127	2,127
cyrus-sasl-plain	2,127	2,127
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
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2

Package	AMI	VMware OVA
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
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188

Package	AMI	VMware OVA
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
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

Package	AMI	VMware OVA
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
grub2-pc		2,06
grub2-pc-modules	2,06	2,06
grub2-tools	2,06	2,06
grub2-tools-minimal	2,06	2,06

Package	AMI	VMware OVA
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
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	
json-c	0,14	0,14

Package	AMI	VMware OVA
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6,1,90	6,1,90
kernel-livepatch-repo-cdn		2023,4.20240513
kernel-livepatch-repo-s3	2023,4.20240513	
kernel-modules-extra		6,1,90
kernel-modules-extra-common		6,1,90
kernel-srpm-macros	1.0	1.0
kernel-tools	6,1,90	6,1,90
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

Package	AMI	VMware OVA
libarchive	3.5.3	3.5.3
libargon2	27/12/2017	27/12/2017
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0,11	0,11
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
libeconf	0,4,0	0,4,0
libedit	3.1	3.1
libev	4,33	4,33

Package	AMI	VMware OVA
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	0,13	0,13
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2,37,4	2,37,4
libmspack		0,1,1

Package	AMI	VMware OVA
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	0,15	0,15
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

Package	AMI	VMware OVA
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
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

Package	AMI	VMware OVA
libverto-libev	0,3.2	0,3.2
libxcrypt	4.4.33	4.4.33
libxml2	2.10.4	2.10.4
libxslt		1,134
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
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

Package	AMI	VMware OVA
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	8,7 p1	8,7 p1
openssh-clients	8,7 p1	8,7 p1

Package	AMI	VMware OVA
openssh-server	8,7 p1	8,7 p1
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	1,77	1,77
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	10,40	10,40
pcre2-syntax	10,40	10,40
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

Package	AMI	VMware OVA
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

Package	AMI	VMware OVA
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	2021,0726	2021,0726
perl-Time-Local	1,300	1,300

Package	AMI	VMware OVA
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	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

Package	AMI	VMware OVA
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

Package	AMI	VMware OVA
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	0,1,6	0,1,6
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

Package	AMI	VMware OVA
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	0,12	0,12
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

Package	AMI	VMware OVA
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	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

Package	AMI	VMware OVA
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	252,16	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
systemd-udev	252,16	252,16
system-release	2023,4.20240513	2023,4.20240513
systemtap-runtime	4.8	4.8
tar	1,34	1,34
tbb	2020,3	2020,3
tcpdump	4,99,1	4,99,1
tcsh	6,24,07	6,24,07
time	1.9	1.9

Package	AMI	VMware OVA
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
xmlsec1		1.2.33
xmlsec1-openssl		1.2.33
xxd	9,0.2153	9,0.2153

Package	AMI	VMware OVA
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

Comparaison des packages installés sur l'AMI standard Amazon Linux 2023 avec l'image Hyper-V AL2023

Comparaison des RPM présents sur l'AMI standard AL2023 par rapport aux RPM présents sur l'image Hyper-V AL2023.

Package	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	

Package	AMI	Hyper-V VHDX
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
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

Package	AMI	Hyper-V VHDX
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
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

Package	AMI	Hyper-V VHDX
cracklib-dicts	2,9,6	2,9,6
crontabs	1.11	1.11
crypto-policies	20220428	20220428
crypto-policies-scripts	20220428	20220428
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	2,127	2,127
cyrus-sasl-plain	2,127	2,127
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

Package	AMI	Hyper-V VHDX
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	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5

Package	AMI	Hyper-V VHDX
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
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

Package	AMI	Hyper-V VHDX
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
grub2-pc		2,06
grub2-pc-modules	2,06	2,06

Package	AMI	Hyper-V VHDX
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
hyperv-tools		0
hypervvssd		0
info	6.7	6.7
inih	49	49

Package	AMI	Hyper-V VHDX
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	6,179	6,179
kernel-livepatch-repo-cdn		2023,4.20240319
kernel-livepatch-repo-s3	2023,4.20240319	
kernel-modules-extra		6,179
kernel-modules-extra-common		6,179
kernel-srpm-macros	1.0	1.0
kernel-tools	6,179	6,179
keyutils	1.6.3	1.6.3

Package	AMI	Hyper-V VHDX
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	27/12/2017	27/12/2017
libassuan	2.5.5	2.5.5
libattr	2.5.1	2.5.1
libbasicobjects	0,11	0,11
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

Package	AMI	Hyper-V VHDX
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
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

Package	AMI	Hyper-V VHDX
libkcapi-hmaccalc	1.4.0	1.4.0
libldb	2.6.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0,13	0,13
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
libref_array	0,15	0,15
librepo	1.14,5	1.14,5
libreport-filesystem	2.15.2	2.15.2
libseccomp	2.5.3	2.5.3

Package	AMI	Hyper-V VHDX
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
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0,21	0,21
libtirpc	1.3.3	1.3.3

Package	AMI	Hyper-V VHDX
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	4.4.33	4.4.33
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
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

Package	AMI	Hyper-V VHDX
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
numactl-libs	2,0,14	2,0,14
ocaml-srpm-macros	6	6

Package	AMI	Hyper-V VHDX
oniguruma	6.9.7.1	6.9.7.1
openblas-srpm-macros	2	2
openldap	2,4,57	2,4,57
openssh	8,7 p1	8,7 p1
openssh-clients	8,7 p1	8,7 p1
openssh-server	8,7 p1	8,7 p1
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	1,77	1,77
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	10,40	10,40
pcre2-syntax	10,40	10,40

Package	AMI	Hyper-V VHDX
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
perl-IO	1,43	1,43
perl-IPC-Open3	1,21	1,21
perl-libs	5,32.1	5,32.1

Package	AMI	Hyper-V VHDX
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
perl-subs	1,03	1,03
perl-Symbol	1,08	1,08
perl-Term-ANSIColor	5,01	5,01

Package	AMI	Hyper-V VHDX
perl-Term-Cap	1,17	1,17
perl-Text-ParseWords	3,30	3,30
perl-Text-Tabs+Wrap	2021,0726	2021,0726
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	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

Package	AMI	Hyper-V VHDX
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
python3-jinja2	2.11.3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1,21	1,21

Package	AMI	Hyper-V VHDX
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	0,1,6	0,1,6
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

Package	AMI	Hyper-V VHDX
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	0,12	0,12
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

Package	AMI	Hyper-V VHDX
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	37,22	37,22
selinux-policy-targeted	37,22	37,22
setup	2.13.7	2.13.7

Package	AMI	Hyper-V VHDX
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
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

Package	AMI	Hyper-V VHDX
tbb	2020,3	2020,3
tcpdump	4,99,1	4,99,1
tcsh	6,24,07	6,24,07
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
words	3.0	3.0
xfsdump	3.1.11	3.1.11

Package	AMI	Hyper-V VHDX
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

Mise à jour d'AL2023

Il est important de rester à jour avec les versions d'AL2023 afin de bénéficier des mises à jour de sécurité et des nouvelles fonctionnalités. Avec AL2023, vous pouvez garantir la cohérence entre les versions des packages et les mises à jour dans votre environnement avec Utilisation de mises à niveau déterministes via un référentiel versionné sur AL2023.

Rubriques

Recevez des notifications sur les nouvelles mises à jour

Vous pouvez recevoir des notifications chaque fois qu'une nouvelle AMI AL2023 est publiée. Les notifications sont publiées sur Amazon SNS à l'aide de la rubrique suivante.

```
arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates
```

Des messages sont publiés ici lorsqu'une nouvelle AMI AL2023 est publiée. La version de l'AMI est incluse dans le message.

Ces messages peuvent être reçus à l'aide de différentes méthodes. Nous vous recommandons d'utiliser la méthode suivante.

1. Ouvrez la console Amazon SNS.
2. Dans la barre de navigation, remplacez le Région AWS par US East (Virginie du Nord), si nécessaire. Vous devez sélectionner la région dans laquelle la notification SNS à laquelle vous vous abonnez a été créée.
3. Dans le panneau de navigation, choisissez Abonnements, puis Créer un abonnement.
4. Dans la boîte de dialogue Créer un abonnement, procédez comme suit :
   a. Pour l'ARN du sujet, copiez et collez le nom de ressource Amazon (ARN) suivant :**`arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates`**.

b. Pour Protocole, choisissez E-mail.

c. Pour Point de terminaison, entrez une adresse e-mail que vous pouvez utiliser pour recevoir les notifications.

d. Choisissez Créer un abonnement.

5. Vous recevez un e-mail de confirmation dont l'objet est « AWS Notification - Confirmation d'abonnement ». Ouvrez l'e-mail et choisissez Confirm subscription (Confirmer l'abonnement) pour terminer votre abonnement.

Gérez les mises à jour des packages et du système d'exploitation dans AL2023

Contrairement aux versions précédentes d'Amazon Linux, les AMI AL2023 sont verrouillées sur une version spécifique du référentiel Amazon Linux. Pour appliquer à la fois des correctifs de sécurité et de bogues à une instance AL2023, mettez à jour la configuration DNF. Vous pouvez également lancer une instance AL2023 plus récente.

Cette section décrit comment gérer les packages et les référentiels DNF sur une instance en cours d'exécution. Il décrit également comment configurer DNF à partir d'un script de données utilisateur pour activer le dernier référentiel Amazon Linux disponible au moment du lancement. Pour plus d'informations, consultez Référence de la commande DNF.

Rubriques

Vérification des mises à jour de package disponibles

Vous pouvez utiliser cette commande `dnf check-update` pour vérifier les mises à jour éventuelles de votre système. Pour AL2023, nous vous recommandons d'ajouter l'option `--releasever=`*`version-number`* à la commande.

Lorsque vous ajoutez cette option, DNF vérifie également les mises à jour pour une version ultérieure du référentiel. Par exemple, après avoir exécuté la commande `dnf check-update`, utilisez la dernière version renvoyée comme valeur pour *`version-number`*.

Si l'instance est mise à jour pour utiliser la dernière version du référentiel, la sortie inclut une liste de tous les packages à mettre à jour.

> **Note**
>
> Si vous ne spécifiez pas la version de publication avec l'indicateur facultatif associé à la commande `dnf check-update`, seule la version du référentiel actuellement configurée est vérifiée. Cela signifie que les packages de la dernière version du référentiel ne sont pas vérifiés.

```
$ sudo dnf check-update --releasever=2023.0.20230210
Last metadata expiration check: 0:06:13 ago on Mon 13 Feb 2023 10:39:32 PM UTC.

bind-libs.x86_64                        32:9.16.27-1.amzn2023          amazonlinux
bind-license.noarch                     32:9.16.27-1.amzn2023          amazonlinux
bind-utils.x86_64                       32:9.16.27-1.amzn2023          amazonlinux
cloud-init.noarch                       22.2.2-1.amzn2023.1.4          amazonlinux
dnf.noarch                              4.12.0-2.amzn2023.0.1          amazonlinux
dnf-data.noarch                         4.12.0-2.amzn2023.0.1          amazonlinux
dracut.x86_64                           055-6.amzn2023.0.4             amazonlinux
dracut-config-generic.x86_64            055-6.amzn2023.0.4             amazonlinux
glib2.x86_64                            2.73.2-678.amzn2023            amazonlinux
gmp.x86_64                              1:6.2.1-2.amzn2023             amazonlinux
grep.x86_64                             3.8-1.amzn2023.0.1             amazonlinux
kpatch-runtime.noarch                   0.9.4-7.amzn2023               amazonlinux
libgcc.x86_64                           11.3.1-2.amzn2023.0.6          amazonlinux
libgomp.x86_64                          11.3.1-2.amzn2023.0.6          amazonlinux
libpkgconf.x86_64                       1.7.3-7.amzn2023.0.1           amazonlinux
libstdc++.x86_64                        11.3.1-2.amzn2023.0.6          amazonlinux
```

```
lz4-libs.x86_64                     1.9.4-1.amzn2023              amazonlinux
pkgconf.x86_64                      1.7.3-7.amzn2023.0.1          amazonlinux
pkgconf-m4.noarch                   1.7.3-7.amzn2023.0.1          amazonlinux
pkgconf-pkg-config.x86_64           1.7.3-7.amzn2023.0.1          amazonlinux
python3-dnf.noarch                  4.12.0-2.amzn2023.0.1         amazonlinux
python3-rpm.x86_64                  4.16.1.3-12.amzn2023.0.2      amazonlinux
rpm.x86_64                          4.16.1.3-12.amzn2023.0.2      amazonlinux
rpm-build-libs.x86_64               4.16.1.3-12.amzn2023.0.2      amazonlinux
rpm-libs.x86_64                     4.16.1.3-12.amzn2023.0.2      amazonlinux
rpm-plugin-selinux.x86_64           4.16.1.3-12.amzn2023.0.2      amazonlinux
rpm-plugin-systemd-inhibit.x86_64   4.16.1.3-12.amzn2023.0.2      amazonlinux
rpm-sign-libs.x86_64                4.16.1.3-12.amzn2023.0.2      amazonlinux
slang.x86_64                        2.3.2-9.amzn2023.0.1          amazonlinux
system-release.noarch               2023.0.20230210-0.amzn2023    amazonlinux
systemd.x86_64                      250.8-1.amzn2023.0.1          amazonlinux
systemd-libs.x86_64                 250.8-1.amzn2023.0.1          amazonlinux
systemd-networkd.x86_64             250.8-1.amzn2023.0.1          amazonlinux
systemd-pam.x86_64                  250.8-1.amzn2023.0.1          amazonlinux
systemd-resolved.x86_64             250.8-1.amzn2023.0.1          amazonlinux
systemd-udev.x86_64                 250.8-1.amzn2023.0.1          amazonlinux
vim-common.x86_64                   2:9.0.327-1.amzn2023.0.1      amazonlinux
vim-data.noarch                     2:9.0.327-1.amzn2023.0.1      amazonlinux
vim-enhanced.x86_64                 2:9.0.327-1.amzn2023.0.1      amazonlinux
vim-filesystem.noarch               2:9.0.327-1.amzn2023.0.1      amazonlinux
vim-minimal.x86_64                  2:9.0.327-1.amzn2023.0.1      amazonlinux
wget.x86_64                         1.21.3-1.amzn2023             amazonlinux
yum.noarch                          4.12.0-2.amzn2023.0.1         amazonlinux
```

Pour cette commande, si de nouveaux packages sont disponibles, le code de retour est 100. Si aucun nouveau package n'est disponible, le code de retour est 0. En outre, le résultat répertorie également tous les packages à mettre à jour.

Application des mises à jour de sécurité à l'aide du DNF et des versions du référentiel

Les nouvelles mises à jour de package et les mises à jour de sécurité ne sont disponibles que pour les nouvelles versions du référentiel. Pour les instances que vous avez lancées à partir de versions antérieures de l'AMI AL2023, vous devez mettre à jour la version du référentiel avant de pouvoir installer les mises à jour de sécurité. La commande `dnf check-release-update` inclut un exemple de commande de mise à jour qui met à jour tous les packages installés sur le système vers les versions d'un référentiel plus récent.

```
$ sudo dnf update --releasever=2023.0.20230210
Last metadata expiration check: 0:01:40 ago on Mon 13 Feb 2023 10:39:32 PM UTC.
Dependencies resolved.
================================================================================
 Package                     Arch   Version                     Repository   Size
================================================================================
Upgrading:
 bind-libs                   x86_64 32:9.16.27-1.amzn2023      amazonlinux 1.2 M
 bind-license                noarch 32:9.16.27-1.amzn2023      amazonlinux  16 k
 bind-utils                  x86_64 32:9.16.27-1.amzn2023      amazonlinux 202 k
 cloud-init                  noarch 22.2.2-1.amzn2023.1.4      amazonlinux 1.1 M
 dnf                         noarch 4.12.0-2.amzn2023.0.1      amazonlinux 454 k
 dnf-data                    noarch 4.12.0-2.amzn2023.0.1      amazonlinux  42 k
 dracut                      x86_64 055-6.amzn2023.0.4         amazonlinux 345 k
 dracut-config-generic       x86_64 055-6.amzn2023.0.4         amazonlinux 8.5 k
 glib2                       x86_64 2.73.2-678.amzn2023        amazonlinux 2.7 M
 gmp                         x86_64 1:6.2.1-2.amzn2023         amazonlinux 324 k
 grep                        x86_64 3.8-1.amzn2023.0.1         amazonlinux 316 k
 kpatch-runtime              noarch 0.9.4-7.amzn2023           amazonlinux  30 k
 libgcc                      x86_64 11.3.1-2.amzn2023.0.6      amazonlinux 121 k
 libgomp                     x86_64 11.3.1-2.amzn2023.0.6      amazonlinux 296 k
 libpkgconf                  x86_64 1.7.3-7.amzn2023.0.1       amazonlinux  37 k
 libstdc++                   x86_64 11.3.1-2.amzn2023.0.6      amazonlinux 758 k
 lz4-libs                    x86_64 1.9.4-1.amzn2023           amazonlinux  81 k
 pkgconf                     x86_64 1.7.3-7.amzn2023.0.1       amazonlinux  41 k
 pkgconf-m4                  noarch 1.7.3-7.amzn2023.0.1       amazonlinux  15 k
 pkgconf-pkg-config          x86_64 1.7.3-7.amzn2023.0.1       amazonlinux  11 k
 python3-dnf                 noarch 4.12.0-2.amzn2023.0.1      amazonlinux 415 k
 python3-rpm                 x86_64 4.16.1.3-12.amzn2023.0.2   amazonlinux  89 k
 rpm                         x86_64 4.16.1.3-12.amzn2023.0.2   amazonlinux 487 k
 rpm-build-libs              x86_64 4.16.1.3-12.amzn2023.0.2   amazonlinux  92 k
 rpm-libs                    x86_64 4.16.1.3-12.amzn2023.0.2   amazonlinux 311 k
 rpm-plugin-selinux          x86_64 4.16.1.3-12.amzn2023.0.2   amazonlinux  18 k
 rpm-plugin-systemd-inhibit x86_64 4.16.1.3-12.amzn2023.0.2   amazonlinux  19 k
 rpm-sign-libs               x86_64 4.16.1.3-12.amzn2023.0.2   amazonlinux  22 k
 slang                       x86_64 2.3.2-9.amzn2023.0.1       amazonlinux 410 k
 system-release              noarch 2023.0.20230210-0.amzn2023 amazonlinux  25 k
 systemd                     x86_64 250.8-1.amzn2023.0.1       amazonlinux 4.2 M
 systemd-libs                x86_64 250.8-1.amzn2023.0.1       amazonlinux 615 k
 systemd-networkd            x86_64 250.8-1.amzn2023.0.1       amazonlinux 614 k
 systemd-pam                 x86_64 250.8-1.amzn2023.0.1       amazonlinux 335 k
 systemd-resolved            x86_64 250.8-1.amzn2023.0.1       amazonlinux 277 k
 systemd-udev                x86_64 250.8-1.amzn2023.0.1       amazonlinux 1.9 M
```

```
 vim-common                 x86_64 2:9.0.327-1.amzn2023.0.1   amazonlinux 7.2 M
 vim-data                   noarch 2:9.0.327-1.amzn2023.0.1   amazonlinux  27 k
 vim-enhanced               x86_64 2:9.0.327-1.amzn2023.0.1   amazonlinux 1.8 M
 vim-filesystem             noarch 2:9.0.327-1.amzn2023.0.1   amazonlinux  21 k
 vim-minimal                x86_64 2:9.0.327-1.amzn2023.0.1   amazonlinux 764 k
 wget                       x86_64 1.21.3-1.amzn2023          amazonlinux 813 k
 yum                        noarch 4.12.0-2.amzn2023.0.1      amazonlinux  39 k

Transaction Summary
================================================================================
Upgrade  43 Packages
...
```

Vous pouvez ajouter l'option `--security` pour mettre à jour les packages uniquement avec des fonctionnalités de sécurité.

```
$ sudo dnf update --releasever=2023.0.20230210 --security
Amazon Linux 2023 repository                    18 MB/s |  11 MB     00:00
Last metadata expiration check: 0:00:02 ago on Mon 13 Feb 2023 10:39:32 PM UTC.
Dependencies resolved.
================================================================================
 Package           Arch      Version                        Repository      Size
================================================================================
Upgrading:
 bind-libs         x86_64    32:9.16.27-1.amzn2023          amazonlinux    1.2 M
 bind-license      noarch    32:9.16.27-1.amzn2023          amazonlinux     16 k
 bind-utils        x86_64    32:9.16.27-1.amzn2023          amazonlinux    202 k
 gmp               x86_64    1:6.2.1-2.amzn2023             amazonlinux    324 k
 lz4-libs          x86_64    1.9.4-1.amzn2023               amazonlinux     81 k
 vim-common        x86_64    2:9.0.327-1.amzn2023.0.1       amazonlinux    7.2 M
 vim-data          noarch    2:9.0.327-1.amzn2023.0.1       amazonlinux     27 k
 vim-enhanced      x86_64    2:9.0.327-1.amzn2023.0.1       amazonlinux    1.8 M
 vim-filesystem    noarch    2:9.0.327-1.amzn2023.0.1       amazonlinux     21 k
 vim-minimal       x86_64    2:9.0.327-1.amzn2023.0.1       amazonlinux    764 k
 wget              x86_64    1.21.3-1.amzn2023              amazonlinux    813 k

Transaction Summary
================================================================================
Upgrade  11 Packages
...
```

Pour découvrir les versions du package AL2023, effectuez une ou plusieurs des opérations suivantes :

- Exécutez la commande `dnf check-update`.
- Abonnez-vous à la rubrique SNS relative à la mise à jour du référentiel Amazon Linux (`arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates`). Pour plus d'informations, consultez Abonnement à une rubrique Amazon SNS dans le Guide du développeur Amazon Simple Notification Service.
- Consultez régulièrement les notes de publication d'AL2023.

> **Important**
>
> Lorsque vous appliquez des mises à jour de sécurité à une instance en cours d'exécution, assurez-vous que le DNF pointe vers la dernière version du référentiel.

Redémarrage automatique du service après les mises à jour (de sécurité)

Amazon Linux est désormais livré avec le package smart-restart. `Smart-restart`redémarre les services systemd lors des mises à jour du système chaque fois qu'un package est installé ou supprimé à l'aide du gestionnaire de packages du système. Cela se produit chaque fois qu'`dnf (update|upgrade|downgrade)`il est exécuté.

`Smart-restart`utilise le `needs-restarting` package from `dnf-utils` et un mécanisme de refus personnalisé pour déterminer quels services doivent être redémarrés et s'il est conseillé de redémarrer le système. Si un redémarrage du système est conseillé, un fichier de marqueurs de redémarrage est généré (`/run/smart-restart/reboot-hint-marker`).

Pour installer **`smart-restart`**

Exécutez la DNF commande suivante (comme vous le feriez avec n'importe quel autre package).

```
$ sudo dnf install smart-restart
```

Après l'installation, les transactions suivantes déclencheront la `smart-restart` logique.

Liste de refus

`Smart-restart`peut être invité à bloquer le redémarrage de certains services. Les services bloqués ne contribueront pas à déterminer si un redémarrage est nécessaire. Pour bloquer des services

supplémentaires, ajoutez un fichier avec le suffixe « `-denylist` in »`/etc/smart-restart-conf.d/`, comme indiqué dans l'exemple suivant.

```
$ cat /etc/smart-restart-conf.d/custom-denylist
# Some comments
myservice.service
```

> **Note**
>
> Tous les `*-denylist` fichiers sont lus et évalués au moment de décider si un redémarrage est nécessaire.

Crochets personnalisés

Outre le denylisting, `smart-restart` fournit un mécanisme permettant d'exécuter des scripts personnalisés avant et après les tentatives de redémarrage du service. Les scripts personnalisés peuvent être utilisés pour effectuer manuellement des étapes de préparation ou pour informer les autres composants d'un redémarrage restant ou terminé.

Tous les scripts `/etc/smart-restart-conf.d/` dotés du suffixe `-pre-restart` ou `-post-restart` sont exécutés. Si l'ordre est important, préfixez tous les scripts par un numéro pour garantir l'ordre d'exécution, comme indiqué dans l'exemple suivant.

```
$ ls /etc/smart-restart-conf.d/*-pre-restart
001-my-script-pre-restart
002-some-other-script-pre-restart
```

Lancement d'une instance avec la dernière version du référentiel activée

Vous pouvez ajouter des commandes DNF à un script de données utilisateur pour contrôler les packages RPM installés sur une AMI Amazon Linux lors de son lancement. Dans l'exemple suivant, un script de données utilisateur est utilisé pour s'assurer que les mêmes mises à jour de package sont installées sur toute instance lancée avec le script de données utilisateur.

```
#!/bin/bash
dnf update --releasever=2023.0.20230210
# Additional setup and install commands below
```

```
dnf install httpd php7.4 mysql80
```

Vous devez exécuter ce script en tant que superutilisateur (racine). Pour ce faire, exécutez la commande suivante.

```
$ sudo sh -c "bash nameofscript.sh"
```

Pour plus d'informations, consultez la section Données utilisateur et scripts shell dans le guide de l'utilisateur Amazon EC2.

> **Note**
>
> Au lieu d'utiliser un script de données utilisateur, lancez la dernière AMI Amazon Linux ou une AMI personnalisée basée sur l'AMI Amazon Linux. La dernière AMI Amazon Linux possède toutes les mises à jour nécessaires installées et est configurée pour pointer vers une version de référentiel particulière.

Obtention des informations de support relatives aux packages

AL2023 intègre de nombreux projets de logiciels open source différents. Chacun de ces projets est géré indépendamment d'Amazon Linux et possède des versions et des end-of-support calendriers différents. Pour vous fournir des informations spécifiques à Amazon Linux sur ces différents packages, le plug-in DNF `supportinfo` fournit des métadonnées relatives à un package. Dans l'exemple suivant, la commande **`dnf supportinfo`** renvoie les métadonnées du package `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.
```

Vérification des nouvelles versions du référentiel

Dans une instance AL2023, vous pouvez utiliser l'utilitaire DNF pour gérer des référentiels et appliquer des packages RPM mis à jour. Ces packages sont disponibles dans les référentiels Amazon Linux. Vous pouvez utiliser la commande DNF `dnf check-release-update` pour vérifier les nouvelles versions du référentiel DNF.

```
$ 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 update --releasever=2023.0.20230210

    Release notes:
     https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes.html
```

Cela renvoie une liste complète de toutes les nouvelles versions des référentiels DNF disponibles. Si rien n'est renvoyé, cela signifie que DNF est actuellement configuré pour utiliser la dernière version disponible. La version du package `system-release` actuellement installé définit la variable `releasever` DNF. Pour vérifier la version actuelle du référentiel, exécutez la commande suivante.

```
$ rpm -q system-release --qf "%{VERSION}\n"
```

Lorsque vous exécutez des transactions de package DNF (telles que des commandes d'installation, de mise à jour ou de suppression), un message d'avertissement vous informe de toute nouvelle version du référentiel. Par exemple, si vous installez le package `httpd` sur une instance lancée à partir d'une ancienne version d'AL2023, le résultat suivant est renvoyé.

```
$ 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!
```

Ajout, activation ou désactivation de nouveaux référentiels

Pour installer un package d'un référentiel différent avec la commande DNF, vous devez ajouter les détails relatifs au référentiel au fichier `/etc/dnf/dnf.conf` ou à son propre fichier *`repository.repo`* dans le répertoire `/etc/yum.repos.d`. Vous pouvez effectuer cela manuellement. Cependant, la plupart des référentiels DNF ont leur propre fichier *`repository.repo`* sur l'URL de leur référentiel.

> **Note**
>
> À l'heure actuelle, aucun référentiel supplémentaire ne peut être ajouté à AL2023. Cela peut changer à l'avenir. Vous pouvez également écrire vos propres packages et les mettre à disposition de votre environnement d'entreprise AL2023. Avant de pouvoir utiliser les packages, vous devez ajouter et activer le référentiel dans lequel les packages sont stockés.

Pour savoir quels référentiels sont actuellement activés, vous pouvez exécuter la commande suivante :

```
$ 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

Si vous n'ajoutez pas l'indicateur d'option `--verbose`, la sortie inclut uniquement les informations `Repo-id`, `Repo-name` et `Repo-status`.

Pour ajouter un référentiel **yum** à un répertoire **/etc/yum.repos.d** :

1. Recherchez l'emplacement du fichier `.repo`. Dans cet exemple, le fichier `.repo` se trouve à l'adresse *https://www.example.com/repository.repo*.
2. Ajoutez le référentiel à l'aide de la commande `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
```

Après avoir installé un référentiel, vous devez l'activer, comme décrit dans la procédure suivante.

Pour activer un référentiel yum dans `/etc/yum.repos.d`, utilisez la commande `dnf config-manager` avec l'indicateur `--enable` et le nom du *référentiel*.

```
$ sudo dnf config-manager --enable repository
```

> **Note**
>
> Pour désactiver un référentiel, utilisez la même syntaxe de commande, mais remplacez `--enable` par `--disable` dans la commande.

Ajout de référentiels avec cloud-init

Outre l'ajout d'un référentiel à l'aide de la méthode précédente, vous pouvez également ajouter un nouveau référentiel à l'aide du framework `cloud-init`.

Pour ajouter un nouveau référentiel de package, nous vous recommandons d'utiliser le modèle suivant. Envisagez d'enregistrer ce fichier localement.

```
#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**
>
> L'un des avantages de l'utilisation de `cloud-init` est que vous pouvez ajouter une section `packages:` à votre fichier de configuration. Dans cette section, vous pouvez inclure les noms des packages que vous souhaitez installer. Vous pouvez installer des packages à partir du référentiel par défaut ou du nouveau référentiel que vous avez ajouté dans le fichier `cloud-config`.
>
> Pour des informations plus spécifiques sur la structure du fichier YAML, consultez Ajout d'un référentiel YUM dans la documentation *`cloud-init`*.

Après avoir configuré le fichier au format YAML, vous pouvez l'exécuter dans le framework `cloud-init` d' AWS CLI. Assurez-vous d'inclure l'option `--userdata` et le nom du fichier `.yml` pour appeler les opérations souhaitées.

```
$ 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
```

Utilisation de mises à niveau déterministes via un référentiel versionné sur AL2023

> **Note**
>
> Par défaut, votre instance AL2023 ne reçoit pas automatiquement de mises à jour de sécurité critiques et importantes supplémentaires au lancement. L'instance contient initialement les mises à jour disponibles dans la version d'AL2023 et l'AMI sélectionnée.

Contrôle des mises à jour reçues à partir des versions majeures et mineures

Avec AL2023, vous pouvez garantir la cohérence entre les versions des packages et les mises à jour dans votre environnement. Vous pouvez également garantir la cohérence de plusieurs instances d'une même Amazon Machine Image (AMI). Grâce à la fonctionnalité de mises à niveau déterministes via des référentiels versionnés, qui est activée par défaut, vous pouvez appliquer des mises à jour selon un calendrier qui répond à vos besoins spécifiques.

Chaque fois que nous publions de nouvelles mises à jour de packages, il existe une nouvelle version à verrouiller et de nouvelles AMI verrouillées sur cette version.

AL2023 se verrouille sur une version spécifique de votre référentiel. Ceci est pris en charge pour les versions majeures ou mineures. L'AMI AL2023, exposée via les paramètres SSM, est toujours la dernière version. Il contient le plus grand nombre de up-to-date packages et de mises à jour, y compris les mises à jour de sécurité critiques et importantes.

Si vous lancez une instance à partir d'une AMI existante, les mises à jour ne sont pas appliquées automatiquement. Tous les packages supplémentaires installés dans le cadre du provisionnement correspondent à la version du référentiel de l'AMI existante.

Grâce à cette fonctionnalité, vous êtes chargé de garantir la cohérence entre les versions des packages et les mises à jour dans votre environnement. Cela est particulièrement le cas si vous lancez plusieurs instances à partir de la même AMI. Vous pouvez appliquer des mises à jour en fonction d'un calendrier qui répond à vos besoins. Vous pouvez également appliquer un ensemble spécifique de mises à jour au lancement, car celles-ci peuvent également être verrouillées sur une version de référentiel spécifique.

Différences entre les mises à niveau des versions majeures et mineures

Les versions majeures d'AL2023 incluent des mises à jour à grande échelle et peuvent ajouter, supprimer ou mettre à jour des packages. Pour garantir la compatibilité, ne mettez à niveau votre instance vers une nouvelle version majeure qu'après avoir testé votre application sur cette version.

Les versions mineures d'AL2023 incluent des mises à jour de fonctionnalités et de sécurité, mais n'incluent pas de modifications des packages. Cela garantit que les fonctionnalités Linux et l'API de la bibliothèque système restent disponibles sur les nouvelles versions. Il n'est pas nécessaire de tester votre application avant de procéder à la mise à jour.

Contrôlez les mises à jour des packages disponibles à partir des référentiels AL2023

Lorsque nous publions une nouvelle version des référentiels AL2023, toutes les versions précédentes sont toujours disponibles. Par défaut, le plug-in de gestion des versions des référentiels se verrouille sur la même version que celle utilisée pour créer l'AMI. Si vous souhaitez contrôler les mises à jour des packages, procédez comme suit.

1. Identifiez les versions de référentiel disponibles en exécutant les commandes suivantes.

```
$ sudo dnf check-release-update
```

2. Sélectionnez une version en exécutant la commande suivante.

```
$ sudo dnf --releasever=version update
```

Cette commande lance une mise à jour en utilisant `dnf` de la version actuelle d'Amazon Linux dans la version spécifiée dans la ligne de commande. Une liste des mises à jour de package est présentée par `dnf`. Avant que la mise à jour ne soit traitée, vous devez la confirmer. Une fois la mise à jour terminée, la nouvelle version devient la version par défaut utilisée par `dnf` pour toutes les activités futures.

Pour plus d'informations, voir Gérez les mises à jour des packages et du système d'exploitation dans AL2023.

Mises à niveau déterministes via l'utilisation de référentiels versionnés

Rubriques

- Utilisation d'un système déterministe mis à niveau
- Mise à jour sélective d'un système mis à niveau déterministe
- Utilisation du remplacement persistant avec une mise à niveau déterministe

Utilisation d'un système déterministe mis à niveau

Lorsque vous exécutez la commande `dnf upgrade`, le système vérifie les mises à niveau dans le référentiel spécifié par la variable `releasever`. *Une version valide releasever est soit la plus récente, soit une version horodatée telle que 2023.4.20240513.*

Vous pouvez remplacer cette valeur par `releasever` en utilisant l'une des méthodes suivantes. Ces méthodes sont répertoriées par ordre décroissant de priorité système. Cela signifie que la méthode 1 prévaut sur les méthodes 2 et 3, et que la méthode 2 prévaut sur la méthode 3.

1. Valeur de l'indicateur de ligne de commande, `--releasever=`*`latest`*, s'il est utilisé.
2. Valeur spécifiée dans le fichier de variables de remplacement, `/etc/dnf/vars/releasever`, s'il est défini.
3. Version actuellement installée du package `system-release`.

Dans l'exemple suivant, la version est *`2023.0.20230210`* :

```
$ rpm -q system-release
system-release-2023.0.20230210-0.amzn2023.noarch
```

Dans un système récemment installé, la variable de remplacement n'est pas présente. Aucune mise à niveau n'est disponible, car le système est verrouillé sur la version installée de `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!
```

Vous pouvez obtenir des packages d'une version spécifique en utilisant l'indicateur `releasever` pour fournir la version que vous souhaitez.

```
$ 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
```

Comme l'option `--releasever` remplace les `system-release` et `/etc/dnf/vars/releasever`, le résultat de cette mise à niveau est le suivant :

1. La mise à niveau remplace tous les packages installés qui ont changé entre la version précédente et la nouvelle version.
2. La mise à niveau verrouille le système dans le référentiel de la nouvelle version de `system-release`.

Mise à jour sélective d'un système mis à niveau déterministe

Il peut arriver que vous souhaitiez installer certains packages issus d'une version récente, tout en laissant le système verrouillé sur la version d'origine.

Vous pouvez utiliser `dnf check-update` pour identifier les packages que vous souhaitez mettre à niveau.

```
$ 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
```

Installez les packages que vous souhaitez mettre à niveau. Utilisez `sudo dnf upgrade --releasever=latest` et les noms des packages pour garantir que le package `system-release` reste inchangé.

```
$ 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**
>
> L'utilisation de `sudo dnf upgrade --releasever=latest` met à jour tous les packages, notamment `system-release`. La version restera ainsi verrouillée sur le nouveau `system-release`, sauf si vous définissez le remplacement persistant.

Utilisation du remplacement persistant avec une mise à niveau déterministe

Au lieu d'ajouter `--releasever=latest`, vous pouvez utiliser le remplacement persistant pour déverrouiller le système en définissant la valeur de la variable sur la *`valeur la plus récente`*.

```
$ 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**
>
> Si vous avez utilisé la variable de remplacement `/etc/dnf/vars/releasever`, utilisez la commande suivante pour rétablir le comportement de verrouillage par défaut en effaçant la valeur de remplacement.
>
> ```
> $ sudo rm /etc/dnf/vars/releasever
> ```

Kernel Live Patching sur AL2023

Vous pouvez utiliser Kernel Live Patching for AL2023 pour appliquer des correctifs de failles de sécurité et de bogues critiques à un noyau Linux en cours d'exécution sans redémarrer ni perturber le fonctionnement des applications. Kernel Live Patching peut également améliorer la disponibilité de votre application tout en gardant votre infrastructure sécurisée et à jour.

AWS publie deux types de correctifs dynamiques du noyau pour AL2023 :

- Mises à jour de sécurité : incluent des mises à jour pour les failles et vulnérabilités communes (CVE) Linux. Ces mises à jour sont généralement jugées importantes ou critiques à l'aide des

évaluations Amazon Linux de sécurité. Elles correspondent généralement à un score CVSS (Common Vulnerability Scoring System) égal à 7 ou plus. Dans certains cas, AWS peut fournir des mises à jour avant qu'un CVE ne soit attribué. Dans ces cas, les correctifs peuvent apparaître comme des correctifs de bogues.

- Correctifs de bogues : inclut des correctifs pour les bogues critiques et les problèmes de stabilité qui ne sont pas associés à des CVE.

AWS fournit des correctifs dynamiques du noyau pour une version du noyau AL2023 jusqu'à 3 mois après sa publication. Après cette période, vous devez effectuer une mise à jour vers une version ultérieure du noyau pour continuer à recevoir les correctifs à chaud du noyau.

Les correctifs à chaud du noyau AL2023 sont disponibles sous forme de packages RPM signés dans les référentiels AL2023 existants. Les correctifs peuvent être installés sur des instances individuelles à l'aide des flux de travail du gestionnaire de packages DNF existants. Ils peuvent également être installés sur un groupe d'instances gérées à l'aide de AWS Systems Manager.

Kernel Live Patching sur AL2023 est fourni sans frais supplémentaires.

Rubriques

Limites

Lors de l'application d'un correctif à chaud du noyau, vous ne pouvez pas effectuer de mise en veille prolongée, utiliser des outils de débogage avancés (tels que des outils basés sur `SystemTap`, `kprobes` et `eBPF`) ou accéder aux fichiers de sortie `ftrace` utilisés par l'infrastructure Kernel Live Patching.

Configurations et conditions préalables prises en charge

Kernel Live Patching est pris en charge sur les instances EC2 et les machines virtuelles sur site qui exécutent AL2023.

Pour utiliser Kernel Live Patching sur AL2023, vous devez utiliser :

- Une architecture `x86_64` ou `ARM64` 64 bits
- Noyau version `6.1`

Exigences des stratégies

Pour télécharger des packages depuis les référentiels AL2023, Amazon EC2 doit avoir accès aux compartiments Amazon S3 détenus par le service. Si vous utilisez un point de terminaison Amazon Virtual Private Cloud (VPC) pour Amazon S3 dans votre environnement, assurez-vous que votre politique de point de terminaison VPC autorise l'accès à ces compartiments publics. Le tableau suivant décrit le compartiment Amazon S3 auquel Amazon EC2 peut avoir besoin pour accéder au Kernel Live Patching.

ARN de compartiment S3	Description
*arn:aws:s3 : ::al2023-repos-region -de612dc2/**	Compartiment Amazon S3 contenant des référentiels AL2023

Utiliser l'application Kernel Live Patching

Vous pouvez activer et utiliser Kernel Live Patching sur des instances individuelles à l'aide de la ligne de commande de l'instance elle-même. Vous pouvez également activer et utiliser Kernel Live Patching sur un groupe d'instances gérées à l'aide d' AWS Systems Manager.

Les sections suivantes expliquent comment activer et utiliser Kernel Live Patching sur des instances individuelles à l'aide de la ligne de commande.

Pour plus d'informations sur l'activation et l'utilisation de Kernel Live Patching sur un groupe d'instances gérées, consultez Utilisation de Kernel Live Patching sur les instances AL2023 dans le Guide de l'utilisateur d'AWS Systems Manager .

Rubriques

- Activer Kernel Live Patching
- Afficher les correctifs à chaud du noyau disponibles
- Appliquer des correctifs à chaud du noyau
- Afficher les correctifs à chaud du noyau appliqués

- Désactiver Kernel Live Patching

Activer Kernel Live Patching

Kernel Live Patching est désactivé par défaut sur AL2023. Pour utiliser la correction à chaud, vous devez installer le plug-in DNF pour Kernel Live Patching et activer la fonctionnalité de correction à chaud.

Pour activer Kernel Live Patching

1. Les correctifs à chaud du noyau sont disponibles pour AL2023 avec la version du noyau `6.1`. Pour vérifier la version de votre noyau, exécutez la commande suivante.

   ```
   $ sudo dnf list kernel
   ```

2. Installez le plug-in DNF pour Kernel Live Patching.

   ```
   $ sudo dnf install -y kpatch-dnf
   ```

3. Activez le plug-in DNF pour Kernel Live Patching.

   ```
   $ sudo dnf kernel-livepatch -y auto
   ```

   Cette commande installe également la dernière version du RPM du correctif à chaud du noyau à partir des référentiels configurés.

4. Pour confirmer que le plug-in DNF pour la correction à chaud du noyau a été installé correctement, exécutez la commande suivante.

   Lorsque vous activez Kernel Live Patching, un RPM vide du correctif à chaud du noyau est automatiquement appliqué. Si Kernel Live Patching a été activé avec succès, cette commande renvoie une liste qui inclut le RPM vide initial du correctif à chaud du noyau.

   ```
   $ sudo rpm -qa | grep kernel-livepatch
   dnf-plugin-kernel-livepatch-1.0-0.11.amzn2023.noarch
   kernel-livepatch-6.1.12-17.42-1.0-0.amzn2023.x86_64
   ```

5. Installez le package kpatch.

   ```
   $ sudo dnf install -y kpatch-runtime
   ```

6. Mettez à jour le service kpatch s'il a été installé précédemment.

```
$ sudo dnf update kpatch-runtime
```

7. Démarrez le service kpatch. Ce service charge tous les correctifs à chaud du noyau lors de l'initialisation ou au démarrage.

```
$ sudo systemctl enable kpatch.service && sudo systemctl start kpatch.service
```

Afficher les correctifs à chaud du noyau disponibles

Les alertes de sécurité Amazon Linux sont publiées dans le Centre de sécurité Amazon Linux. Pour plus d'informations sur les alertes de sécurité AL2023, notamment les alertes pour les correctifs à chaud du noyau, consultez le Centre de sécurité Amazon Linux. Les correctifs Kernel Live sont préfixés avec `ALASLIVEPATCH`. Le Centre de sécurité Amazon Linux peut ne pas répertorier les correctifs à chaud du noyau qui corrigent les bogues.

Vous pouvez également découvrir les correctifs à chaud du noyau disponibles pour les avis et les CVE à l'aide de la ligne de commande.

Pour répertorier tous les correctifs à chaud du noyau disponibles pour les avis

Utilisez la commande suivante.

```
$ 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
```

Pour répertorier tous les correctifs à chaud du noyau disponibles pour les CVE

Utilisez la commande suivante de l'.

```
$ 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
```

Appliquer des correctifs à chaud du noyau

Vous appliquez les correctifs à chaud du noyau en utilisant le gestionnaire de packages DNF de la même manière que vous appliquez les mises à jour régulières. Le plug-in DNF pour Kernel Live Patching gère les correctifs à chaud du noyau que vous appliquez et élimine le besoin de redémarrer.

> **Tip**
>
> Nous vous recommandons de mettre à jour le noyau régulièrement à l'aide de Kernel Live Patching pour vous assurer qu'il reste sécurisé et à jour.

Vous pouvez choisir d'appliquer un correctif à chaud du noyau spécifique ou d'appliquer tous les correctifs à chaud du noyau disponibles avec vos mises à jour de sécurité régulières.

Pour appliquer un correctif à chaud du noyau spécifique

1. Obtenez la version du correctif à chaud du noyau à l'aide de l'une des commandes décrites à la section Afficher les correctifs à chaud du noyau disponibles.
2. Appliquez le correctif à chaud du noyau pour le noyau AL2023.

   ```
   $ sudo dnf install kernel-livepatch-kernel_version-package_version.amzn2023.x86_64
   ```

   Par exemple, la commande suivante applique un correctif à chaud du noyau pour la version `6.1.12-17.42` du noyau AL2023.

   ```
   $ sudo dnf install kernel-livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
   ```

Pour appliquer les correctifs à chaud du noyau disponibles avec vos mises à jour de sécurité régulières

Utilisez la commande suivante.

```
$ sudo dnf update --security
```

Omettre l'option `--security`d'inclure les corrections de bogues.

Important

- La version du noyau n'est pas mise à jour après l'application des correctifs à chaud du noyau. La version est mise à jour vers la nouvelle version seulement après le redémarrage de l'instance.
- Un noyau AL2023 reçoit des correctifs à chaud du noyau pendant une période de 3 mois. Une fois cette période écoulée, aucun nouveau correctif à chaud du noyau n'est publié pour cette version du noyau.
- Pour continuer à recevoir les correctifs à chaud du noyau après 3 mois, vous devez redémarrer l'instance pour passer à la nouvelle version du noyau. L'instance continue de recevoir les correctifs à chaud du noyau pendant les 3 mois qui suivent sa mise à jour.
- Pour vérifier la fenêtre de support de votre version du noyau, exécutez la commande suivante :

```
$ sudo dnf kernel-livepatch support
```

Afficher les correctifs à chaud du noyau appliqués

Pour afficher les correctifs à chaud du noyau appliqués

Utilisez la commande suivante.

```
$ 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)
```

La commande renvoie une liste des correctifs à chaud du noyau des mise à jour de sécurité chargés et installés. Voici un exemple de sortie.

Note

Un seul correctif à chaud du noyau peut inclure et installer plusieurs correctifs à chaud.

Désactiver Kernel Live Patching

Si vous n'avez plus besoin d'utiliser Kernel Live Patching, vous pouvez le désactiver à tout moment.

- Désactivez l'utilisation de livepatches :
  1. Désactivez le plug-in :

     ```
     $ sudo dnf kernel-livepatch manual
     ```

  2. Désactivez le service kpatch :

     ```
     $ sudo systemctl disable --now kpatch.service
     ```

- Retirez complètement les outils livepatch :
  1. Supprimez le plug-in :

     ```
     $ sudo dnf remove kpatch-dnf
     ```

  2. Supprimez kpatch-runtime :

     ```
     $ sudo dnf remove kpatch-runtime
     ```

  3. Supprimez tout livepatches installé :

     ```
     $ sudo dnf remove kernel-livepatch\*
     ```

Commencer à programmer des environnements d'exécution sur AL2023

AL2023 fournit différentes versions de certains environnements d'exécution linguistiques. Nous travaillons avec des projets en amont qui prennent en charge plusieurs versions en même temps. Recherchez des informations sur la façon d'installer et de gérer ces packages nommés selon la version à l'aide de la commande `dnf` de recherche et d'installation de ces packages.

Les rubriques suivantes décrivent la façon dont chaque écosystème linguistique existe dans AL2023.

Rubriques

C, C++ et Fortran dans AL2023

AL2023 inclut à la fois la collection de compilateurs GNU (GCC) et le Clang frontend pour LLVM (Low Level Virtual Machine).

La version majeure de GCC restera constante pendant toute la durée de vie d'AL2023. Les versions mineures apportent des corrections de bogues et peuvent être incluses dans les versions AL2023. D'autres correctifs de bogues, de performances et de sécurité pourraient être rétroportés vers la version majeure de GCC livrée avec AL2023.

AL2023 inclut la version 11 de GCC avec les interfaces C (`gcc`), C++ (`g++`) et Fortran (`gfortran`).

AL2023 n'active pas les frontends Ada (`gnat`), Go (`gcc-go`), Objective-C ou Objective-C++.

Les indicateurs de compilateur par défaut avec lesquels les RPM d'AL2023 sont construits incluent des indicateurs d'optimisation et de renforcement. Pour créer votre propre code avec GCC, nous vous recommandons d'inclure des indicateurs d'optimisation et de renforcement.

> **Note**
>
> Quand `gcc --version` est invoqué, une chaîne de version telle que `gcc (GCC) 11.3.1 20221121 (Red Hat 11.3.1-4)` est affichée. `Red Hat` fait référence à la branche de fournisseur GCC sur laquelle le package GCC d'Amazon Linux est basé. Selon l'URL du rapport de bogue indiquée par`gcc --help`, tous les rapports de bogue et les demandes d'assistance doivent être adressés à Amazon Linux.
> Pour plus d'informations sur certains des changements à long terme apportés à cette branche du fournisseur, tels que la `__GNUC_RH_RELEASE__` macro, consultez les sources des packages Fedora.

Pour plus d'informations sur la chaîne d'outils de base, consultezPackages de chaîne d'outils de base glibc, gcc, binutils.

Pour plus d'informations sur AL2023 et sa relation avec les autres distributions Linux, consultezRelation avec Fedora.

Pour plus d'informations sur le changement du triplet du compilateur dans AL2023 par rapport à AL2, voir. Triplet de compilateur

Go dans AL2023

Vous souhaiterez peut-être créer votre propre code écrit Gosur Amazon Linux et utiliser une chaîne d'outils fournie avec AL2023. Comme AL2, AL2023 mettra à jour la Go chaîne d'outils tout au long de la durée de vie du système d'exploitation. Il peut s'agir d'une réponse à des CVE dans la chaîne d'outils que nous expédions, ou dans le cadre d'une publication trimestrielle.

Goest une langue qui évolue relativement rapidement. Il peut arriver que des applications existantes écrites Go doivent s'adapter aux nouvelles versions de la Go chaîne d'outils. Pour plus d'informations surGo, voir Go1 et l'avenir des Go programmes.

Bien que l'AL2023 incorporera de nouvelles versions de la Go chaîne d'outils au cours de sa durée de vie, cela ne sera pas en même temps que les versions en amont. Go Par conséquent, l'utilisation de la Go chaîne d'outils fournie dans AL2023 peut ne pas être appropriée si vous souhaitez créer Go du code en utilisant les fonctionnalités de pointe du Go langage et de la bibliothèque standard.

Pendant la durée de vie d'AL2023, les versions précédentes des packages ne sont pas supprimées des référentiels. Si une ancienne Go chaîne d'outils est requise, vous pouvez choisir de renoncer

aux correctifs de bogues et de sécurité des nouvelles Go chaînes d'outils et d'installer une version précédente à partir des référentiels en utilisant les mêmes mécanismes disponibles pour tous les RPM.

Si vous souhaitez créer votre propre Go code sur AL2023, vous pouvez utiliser la Go chaîne d'outils incluse dans AL2023 en sachant que cette chaîne d'outils pourrait évoluer pendant la durée de vie d'AL2023.

Fonctions Lambda AL2023 écrites en Go

Lorsqu'il Go est compilé en code natif, Lambda le Go traite comme un environnement d'exécution personnalisé. Vous pouvez utiliser le `provided.al2023` runtime pour déployer des Go fonctions sur AL2023 vers Lambda.

Pour plus d'informations, consultez la section Création de fonctions Lambda avec Go dans le Guide du AWS Lambda développeur.

Java dans AL2023

AL2023 fournit plusieurs versions d'Amazon Corretto pour prendre en charge les charges de travail basées. Java Tous les packages Java basés inclus dans AL2023 sont construits avecAmazon Corretto 17 17.

Corretto est une version du kit de développement Open Java (OpenJDK) avec le support à long terme de. Amazon Corretto est certifié à l'aide du kit de compatibilité technique Java (TCK) pour garantir qu'il répond à la norme Java SE et qu'il est disponible sous Linux, et. Windows macOS

Un package Amazon Corretto est disponible pour Corretto 1.8.0, Corretto 11 et Corretto 17.

Chaque version de Corretto figurant dans AL2023 est prise en charge pour la même durée que la version de Corretto, ou jusqu'à la fin de vie d'AL2023, conformément à la première de ces éventualités. Pour plus d'informations, consultez les déclarations de support des packages Amazon Linux et les FAQ Amazon Corretto.

Perl dans AL2023

AL2023 fournit la version 5.32 du Perllangage de programmation.

Bien qu'il Perl ait fourni un haut degré de compatibilité linguistique dans le cadre de Perl 5 versions au cours des dernières décennies, Amazon Linux ne devrait pas passer de la version Perl 5.32 à la

version AL2023. Amazon Linux continuera à appliquer le correctif de sécurité Perl pendant toute la durée de vie d'AL2023, conformément à nos déclarations de support relatives aux packages.

Modules Perl dans AL2023

Différents Perl modules sont conditionnés sous forme de RPM dans AL2023. Bien que de nombreux Perl modules soient disponibles sous forme de RPM, Amazon Linux ne vise pas à empaqueter tous les Perl modules possibles. Les modules fournis sous forme de RPM peuvent être utilisés par d'autres packages RPM de système d'exploitation. Amazon Linux donnera donc la priorité à ces correctifs de sécurité plutôt qu'à de pures mises à jour de fonctionnalités.

AL2023 permet également aux Perl développeurs d'utiliser le gestionnaire de packages idiomatique `CPAN` pour les modules. Perl

PHP dans AL2023

AL2023 fournit actuellement deux versions du langage de PHPprogrammation, chacune supportée pendant la même période qu'en amontPHP. Pour plus d'informations, consultez la section Déclarations de support du Package.

Avec AL2023, vous pouvez utiliser les nouvelles fonctionnalités de la PHP version 8.2, tout en continuant à prendre en charge les applications qui nécessitent la PHP version 8.1.

Migration à partir d'anciennes versions PHP

La PHP communauté en amont a élaboré une documentation de migration complète pour passer de PHP 8.1 à PHP 8.2. Il existe également une documentation sur la migration de PHP 8.0 vers la version 8.1.

AL2 inclut les PHP versions 8.0, 8.1 et 8.2 pour faciliter `amazon-linux-extras` la mise à niveau vers AL2023.

Migration à partir des versions PHP 7.x

> **Note**
>
> Le PHPprojet tient à jour une liste et un calendrier des versions prises en charge, ainsi qu'une liste des branches non prises en charge.

Lors de la sortie d'AL2023, toutes les versions 7.x et 5.x n'PHPétaient pas prises en charge par la PHP communauté et n'étaient pas incluses en tant qu'options dans AL2023.

La PHP communauté en amont a élaboré une documentation de migration complète pour passer de la version PHP 7.4 à la PHP version 8.0. Combinée à la documentation référencée dans la section précédente sur la migration vers les versions PHP 8.1 et PHP 8.2, vous pouvez migrer votre application PHP basée vers une version modernePHP.

Note

AL2 inclut PHP 7,1, 7,2, 7,3 et 7,4 pouces. `amazon-linux-extras` Il est important de noter que tous ces extras sont end-of-life et ne sont pas garantis pour obtenir d'autres mises à jour de sécurité.

Modules PHP dans AL2023

AL2023 inclut de nombreux PHP modules inclus dans PHP Core. L'AL2023 n'a pas pour objectif d'inclure tous les packages de la bibliothèque communautaire d'PHPextensions (PECL).

Python dans AL2023

AL2023 a supprimé la Python version 2.7 et tous les composants Python requis sont désormais écrits pour fonctionner avec Python 3.

AL2023 en met Python 3 à disposition `/usr/bin/python3` afin de maintenir la compatibilité avec le code client, ainsi que le code Python livré avec AL2023, ce chiffre restera sous forme de Python 3.9 pendant toute la durée de vie de l'AL2023.

La version de python vers laquelle `/usr/bin/python3` pointe est considérée comme le système Python et pour AL2023, il s'agit de Python 3.9.

Les nouvelles versions dePython, telles que Python 3.11, sont disponibles sous forme de packages dans AL2023 et sont prises en charge pendant toute la durée de vie des versions en amont. Pour plus d'informations sur la durée de prise en charge de Python 3.11, consultez Python 3.11.

Plusieurs versions de Python peuvent être installées simultanément sur AL2023. Bien que `/usr/bin/python3` ce soit toujours Python 3.9, chaque version de Python possède un espace de noms et

peut être trouvée grâce à son numéro de version. Par exemple, si `python3.11` est installé, `/usr/bin/python3.11` existera aux côtés de `/usr/bin/python3.9` et du lien symbolique `/usr/bin/python3` vers `/usr/bin/python3.9`.

> **Note**
>
> Ne modifiez pas le `/usr/bin/python3` lien symbolique, car cela pourrait endommager les fonctionnalités de base d'AL2023.

Modules Python dans AL2023

Différents Python modules sont conditionnés sous forme de RPM dans AL2023. Généralement, les RPM pour les modules Python seront créés en ciblant uniquement la version système de Python.

Rust dans AL2023

Vous souhaiterez peut-être créer votre code écrit Rustsur Amazon Linux et utiliser une chaîne d'outils fournie avec AL2023.

Comme AL2, AL2023 mettra à jour la Rust chaîne d'outils tout au long de la durée de vie du système d'exploitation. Il peut s'agir d'une réponse à des CVE dans la chaîne d'outils que nous expédions, ou dans le cadre d'une publication trimestrielle.

Rust est un langage qui évolue relativement rapidement, avec une nouvelle version quasiment toutes les six semaines. Ces versions peuvent ajouter de nouvelles fonctionnalités de langage ou de bibliothèque standard. Bien que l'AL2023 incorporera de nouvelles versions de la Rust chaîne d'outils au cours de sa durée de vie, cela ne sera pas en même temps que les versions en amont. Rust Par conséquent, l'utilisation de la Rust chaîne d'outils fournie dans AL2023 peut ne pas être appropriée si vous souhaitez créer Rust du code en utilisant les fonctionnalités de pointe du Rust langage.

Pendant la durée de vie d'AL2023, les anciennes versions des packages ne sont pas supprimées des référentiels. Si une ancienne Rust chaîne d'outils est requise, vous pouvez choisir de ne pas corriger les bogues et de sécurité des nouvelles Rust chaînes d'outils et d'installer une ancienne version à partir des référentiels en utilisant les mêmes mécanismes disponibles pour tous les RPM.

Si vous souhaitez créer votre propre Rust code sur AL2023, vous pouvez utiliser la Rust chaîne d'outils incluse dans AL2023 en sachant que cette chaîne d'outils pourrait évoluer pendant la durée de vie d'AL2023.

Fonctions Lambda AL2023 écrites en Rust

Comme il Rust compile en code natif, Lambda le Rust traite comme un environnement d'exécution personnalisé. Vous pouvez utiliser le `provided.al2023` runtime pour déployer des Rust fonctions sur AL2023 vers Lambda.

Pour plus d'informations, consultez la section Création de fonctions Lambda avec Rust dans le Guide du AWS Lambda développeur.

Sécurité et conformité dans Amazon Linux 2

> **Important**
>
> Si vous souhaitez signaler une vulnérabilité ou si vous avez un problème de sécurité concernant les services AWS cloud ou les projets open source, contactez le service de AWS sécurité via la page de signalement des vulnérabilités

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des entreprises les plus sensibles en matière de sécurité.

La sécurité est une responsabilité partagée entre vous AWS et vous. Le modèle de responsabilité partagée décrit cela comme la sécurité du cloud et la sécurité dans le cloud :

- Sécurité du cloud : AWS est chargée de protéger l'infrastructure qui exécute les AWS services dans le AWS cloud. AWS vous fournit également des services que vous pouvez utiliser en toute sécurité. Des auditeurs tiers testent et vérifient régulièrement l'efficacité de notre sécurité dans le cadre des programmes de AWS conformité Programmes de de conformité. Pour en savoir plus sur les programmes de conformité qui s'appliquent à AL2, voir AWS Services concernés par programme de conformitéAWS .
- Sécurité dans le cloud : votre responsabilité est déterminée par le AWS service que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris la sensibilité de vos données, les exigences de votre entreprise, ainsi que la législation et la réglementation applicables.

Rubriques

Avis de sécurité d'Amazon Linux pour AL2023

Malgré nos efforts constants pour sécuriser Amazon Linux, des problèmes de sécurité exigeront parfois l'application d'un correctif. Un avis est émis lorsqu'un correctif est disponible. Le principal site où nous publions les avis est le Amazon Linux Security Center (ALAS). Pour plus d'informations, consultez Centre de sécurité Amazon Linux.

> **Important**
>
> Si vous souhaitez signaler une vulnérabilité ou si vous avez un problème de sécurité concernant les services AWS cloud ou les projets open source, contactez le service de AWS sécurité via la page de signalement des vulnérabilités

Les informations sur les problèmes et les mises à jour pertinentes qui affectent AL2023 sont publiées par l'équipe Amazon Linux à plusieurs endroits. Souvent, les outils de sécurité récupèrent ces informations à partir de ces sources principales et vous présentent les résultats. Il se peut donc que vous n'interagissiez pas directement avec les sources principales publiées par Amazon Linux, mais plutôt avec l'interface fournie par votre outil préféré, tel qu'Amazon Inspector.

Annonces du centre de sécurité Amazon Linux

Les annonces Amazon Linux sont fournies pour les articles qui ne rentrent pas dans un avis. Cette section contient des annonces concernant ALAS elle-même, ainsi que des informations qui ne rentrent pas dans un avis. Pour plus d'informations, consultez les annonces du Amazon Linux Security Center (ALAS).

Par exemple, l'annonce 2021-001 - Amazon Linux Hotpatch pour Apache Log4j s'inscrit dans une annonce plutôt que dans un avis. Dans cette annonce, Amazon Linux a ajouté un package destiné à aider les clients à atténuer un problème de sécurité lié à un logiciel qui ne faisait pas partie d'Amazon Linux.

L'explorateur CVE du centre de sécurité Amazon Linux a également été annoncé dans les annonces d'ALAS. Pour plus d'informations, voir Nouveau site Web pour les CVE.

Questions fréquemment posées sur Amazon Linux Security Center

Pour obtenir des réponses aux questions fréquemment posées sur ALAS et sur la façon dont Amazon Linux évalue les CVE, consultez les questions fréquemment posées (FAQ) du Amazon Linux Security Center (ALAS).

Configuration des modes SELinux pour AL2023

Par défaut, Security Enhanced Linux (SELinux) est configuré `enabled` en `permissive` mode AL2023. En mode permissif, les refus d'autorisation sont journalisés mais ne sont pas appliqués. SELinux est un ensemble de fonctionnalités et d'utilitaires du noyau destinés à fournir une architecture de contrôle d'accès obligatoire (MAC) solide, flexible et obligatoire aux principaux sous-systèmes du noyau.

SELinux fournit un mécanisme amélioré pour appliquer la séparation des informations en fonction des exigences de confidentialité et d'intégrité. Cette séparation des informations réduit les risques de falsification et de contournement des mécanismes de sécurité des applications. Elle limite également les dommages pouvant être causés par des applications malveillantes ou défectueuses.

SELinux inclut un ensemble d'exemples de fichiers de configuration des politiques de sécurité conçus pour répondre aux objectifs de sécurité quotidiens.

Pour plus d'informations sur les fonctionnalités de SELinux, consultez SELinux Notebook et Policy Languages.

Rubriques

- État et modes SELinux par défaut pour AL2023
- Passage en mode enforcing
- Option pour désactiver SELinux pour AL2023

État et modes SELinux par défaut pour AL2023

Pour AL2023, SELinux est réglé par défaut sur `enabled` mode. `permissive` En mode `permissive`, les refus d'autorisation sont journalisés mais ne sont pas appliqués.

Les commandes **`getenforce`** ou **`sestatus`** vous indiquent le statut, la politique et le mode actuels de SELinux.

Lorsque le statut par défaut est défini sur `enabled` et `permissive`, la commande **`getenforce`** renvoie `permissive`.

La **`sestatus`** commande renvoie l'état de SELinux et la politique SELinux actuelle, comme indiqué dans l'exemple suivant :

```
$ 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
```

Lorsque vous exécutez SELinux en `permissive` mode, les utilisateurs peuvent mal étiqueter les fichiers. Lorsque vous exécutez SELinux avec le statut `disabled`, les fichiers ne sont pas étiquetés. Les fichiers incorrects ou non étiquetés peuvent entraîner des problèmes lorsque vous passez en mode `enforcing`.

SELinux réétiquette automatiquement les fichiers pour éviter ce problème. SELinux évite les problèmes d'étiquetage grâce au réétiquetage automatique lorsque vous remplacez le statut par `enabled`.

Passage en mode **`enforcing`**

Lorsque vous l'exécutez SELinux en `enforcing` mode, l'SELinuxutilitaire est `enforcing` la politique configurée. SELinuxrégit les capacités de certaines applications en autorisant ou en refusant l'accès conformément aux règles de la politique.

Pour trouver le SELinux mode actuel, exécutez la `getenforce` commande.

```
getenforce
Permissive
```

Modification du fichier de configuration pour activer le mode **`enforcing`**

Pour passer au mode`enforcing`, procédez comme suit.

1. Modifiez le fichier `/etc/selinux/config` pour passer en mode `enforcing`. Le `SELINUX` réglage doit ressembler à l'exemple suivant.

```
SELINUX=enforcing
```

2. Redémarrez le système pour finaliser le passage en mode `enforcing`.

```
$ sudo reboot
```

Au prochain démarrage, SELinux renomme tous les fichiers et répertoires du système. SELinuxajoute également le SELinux contexte pour les fichiers et les répertoires qui ont été créés quand il SELinux a été créé`disabled`.

Après le passage en `enforcing` mode, certaines actions SELinux peuvent être refusées en raison de règles de SELinux politique incorrectes ou manquantes. Vous pouvez consulter les actions refusées à SELinux l'aide de la commande suivante.

```
$ sudo ausearch -m AVC,USER_AVC,SELINUX_ERR,USER_SELINUX_ERR -ts recent
```

Utilisation de cloud-init pour activer le mode **`enforcing`**

Une autre approche pour activer le mode `enforcing` consiste à transmettre la configuration `cloud-config` suivante en tant que données utilisateur lorsque vous lancez l'instance.

```
#cloud-config
selinux:
  mode: enforcing
```

Par défaut, ce paramètre entraîne le redémarrage de l'instance. Pour une plus grande stabilité, nous vous recommandons de redémarrer l'instance. Toutefois, si vous préférez, vous pouvez ignorer le redémarrage en fournissant la configuration `cloud-config` suivante.

```
#cloud-config
selinux:
  mode: enforcing
  selinux_no_reboot: 1
```

Option pour désactiver SELinux pour AL2023

Lorsque vous désactivezSELinux, SELinux la politique n'est ni chargée ni appliquée et les messages du cache vectoriel d'accès (AVC) ne sont pas enregistrés. Vous perdez tous les avantages de la course à piedSELinux.

Au lieu de le désactiverSELinux, nous vous recommandons d'utiliser `permissive` le mode. L'exécution en `permissive` mode coûte un peu plus cher que sa désactivation SELinux complète. Le passage d'un `permissive` `enforcing` mode à un autre nécessite beaucoup moins d'ajustements de configuration que le retour au `enforcing` mode après la désactivation. SELinux Vous pouvez étiqueter les fichiers, tandis que le système peut suivre et journaliser les actions que la politique active aurait pu refuser.

Passer SELinux en **`permissive`** mode

Lorsque vous exécutez SELinux en `permissive` mode, la SELinux politique n'est pas appliquée. En `permissive` mode, SELinux enregistre les messages AVC mais ne refuse pas les opérations. Vous pouvez utiliser ces messages AVC pour le dépannage, le débogage et l'amélioration des SELinux politiques.

Pour SELinux passer en mode permissif, procédez comme suit.

1. Modifiez le fichier `/etc/selinux/config` pour passer en mode `permissive`. La `SELINUX` valeur doit ressembler à l'exemple suivant.

   ```
   SELINUX=permissive
   ```

2. Redémarrez le système pour finaliser le passage en mode `permissive`.

   ```
   sudo reboot
   ```

Désactiver SELinux

Lorsque vous désactivezSELinux, SELinux la politique n'est ni chargée ni appliquée, et les messages AVC ne sont pas enregistrés. Vous perdez tous les avantages de la course à piedSELinux.

Pour le désactiverSELinux, procédez comme suit.

1. Assurez-vous que le `grubby` package est installé.

```
rpm -q grubby
grubby-version
```

2. Configurez le bootloader pour ajouter `selinux=0` à la ligne de commande du noyau.

```
sudo grubby --update-kernel ALL --args selinux=0
```

3. Redémarrez le système.

```
sudo reboot
```

4. Exécutez la `getenforce` commande pour confirmer que c'SELinuxest le cas`Disabled`.

```
$ getenforce
Disabled
```

Pour plus d'informationsSELinux, consultez le SELinuxbloc-notes et SELinuxla configuration.

Activer le mode FIPS sur AL2023

Cette section explique comment activer le mode FIPS (Federal Information Processing Standards) sur AL2023. Pour plus d'informations sur FIPS, consultez les références suivantes :

- Norme FIPS (Federal Information Processing Standard)
- FAQ sur la conformité : normes FIPS

> **Note**
>
> Cette section explique comment activer le mode FIPS dans AL2023. Elle ne couvre pas le statut de certification des modules cryptographiques AL2023.

Prérequis

- Vous devez disposer d'une instance Amazon EC2 AL2023 (AL2023.2 ou version supérieure) avec accès à Internet pour pouvoir télécharger les packages requis. Pour plus d'informations sur le

lancement d'une instance Amazon EC2 AL2023, consultez Lancement d'AL2023 à l'aide de la console Amazon EC2.

- Vous devez vous connecter à votre instance Amazon EC2 en utilisant SSH ou AWS Systems Manager. Pour plus d'informations, consultez Connexion aux instances AL2023.

> **Important**
>
> Les clés utilisateur SSH ED25519 ne sont pas prises en charge en mode FIPS. Si vous avez lancé votre instance Amazon EC2 à l'aide d'une paire de clés SSH ED25519, vous devrez générer de nouvelles clés à l'aide d'un autre algorithme (tel que RSA) ou vous risquez de perdre l'accès à l'instance après l'activation du mode FIPS. Pour plus d'informations, consultez la section Créer des paires de clés dans le guide de l'utilisateur Amazon EC2.

Activation du mode FIPS

1. Connectez-vous à l'instance AL2023 à l'aide de SSH ou AWS Systems Manager.
2. Assurez-vous que le système est à jour. Pour plus d'informations, consultez Gérez les mises à jour des packages et du système d'exploitation dans AL2023.
3. Assurez-vous que les `crypto-policies` utilitaires sont installés et up-to-date.

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

4. Activez le mode FIPS en exécutant la commande suivante.

   ```
   sudo fips-mode-setup --enable
   ```

5. Redémarrez l'instance à l'aide de la commande suivante.

   ```
   sudo reboot
   ```

6. Pour vérifier que le mode FIPS est activé, reconnectez-vous à l'instance et exécutez la commande suivante.

   ```
   sudo fips-mode-setup --check
   ```

   L'exemple de sortie suivant montre que le mode FIPS est activé :

```
FIPS mode is enabled.
Initramfs fips module is enabled.
The current crypto policy (FIPS) is based on the FIPS policy.
```

Durcissement du noyau AL2023

Le noyau Linux 6.1 d'AL2023 est configuré et construit avec plusieurs options et fonctionnalités de renforcement.

Options de sécurisation renforcée du noyau (indépendantes de l'architecture)

Option **CONFIG**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_ACPI_CUSTOM_METHOD	n	n
CONFIG_BINFMT_MISC	m	m
CONFIG_BUG	y	y
CONFIG_BUG_ON_DATA_CORRUPTION	y	y
CONFIG_CFI_CLANG	N/A	N/A
CONFIG_CFI_PERMISSIVE	N/A	N/A
CONFIG_COMPAT	y	y
CONFIG_COMPAT_BRK	n	n
CONFIG_COMPAT_VDSO	N/A	n

Option **CONFIG**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_DEBUG_CREDENTIALS	n	n
CONFIG_DEBUG_LIST	y	y
CONFIG_DEBUG_NOTIFIERS	n	n
CONFIG_DEBUG_SG	n	n
CONFIG_DEBUG_VIRTUAL	n	n
CONFIG_DEBUG_WX	n	n
CONFIG_DEFAULT_MMAP_MIN_ADDR	65536	65536
CONFIG_DEVKMEM	N/A	N/A
CONFIG_DEVMEM	n	n
CONFIG_EFI_DISABLE_PCI_DMA	n	n
CONFIG_FORTIFY_SOURCE	y	y
CONFIG_HARDENED_USERCOPY	y	y
CONFIG_HARDENED_USERCOPY_FALLBACK	N/A	N/A
CONFIG_HARDENED_USERCOPY_PAGESPAN	N/A	N/A
CONFIG_HIBERNATION	y	y

Option **CONFIG**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_HW_RANDOM_TPM	N/A	N/A
CONFIG_INET_DIAG	m	m
CONFIG_INIT_ON_ALLOC_DEFAULT_ON	n	n
CONFIG_INIT_ON_FREE_DEFAULT_ON	n	n
CONFIG_INIT_STACK_ALL_ZERO	N/A	N/A
CONFIG_IOMMU_DEFAULT_DMA_STRICT	n	n
CONFIG_IOMMU_SUPPORT	y	y
CONFIG_IO_STRICT_DEVMEM	N/A	N/A
CONFIG_KEXEC	y	y
CONFIG_KFENCE	n	n
CONFIG_LDISC_AUTOLOAD	n	n
CONFIG_LEGACY_PTYS	n	n
CONFIG_LOCK_DOWN_KERNEL_FORCE_CONFIDENTIALITY	n	n
CONFIG_MODULES	y	y
CONFIG_MODULE_SIG	y	y

Option **CONFIG**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_MODULE_SIG_ALL	y	y
CONFIG_MODULE_SIG_FORCE	n	n
CONFIG_MODULE_SIG_HASH	sha512	sha512
CONFIG_MODULE_SIG_KEY	certs/signing_key.pem	certs/signing_key.pem
CONFIG_MODULE_SIG_SHA512	y	y
CONFIG_PAGE_POISONING	n	n
CONFIG_PAGE_POISONING_NO_SANITY	N/A	N/A
CONFIG_PAGE_POISONING_ZERO	N/A	N/A
CONFIG_PANIC_ON_OOPS	y	y
CONFIG_PANIC_TIMEOUT	0	0
CONFIG_PROC_KCORE	y	y
CONFIG_RANDOMIZE_KSTACK_OFFSET_DEFAULT	n	n
CONFIG_RANDOM_TRUST_BOOTLOADER	y	y

Option **CONFIG**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_RANDOM_TRUST_CPU	y	y
CONFIG_REFCOUNT_FULL	N/A	N/A
CONFIG_SCHED_CORE	N/A	y
CONFIG_SCHED_STACK_END_CHECK	y	y
CONFIG_SECCOMP	y	y
CONFIG_SECCOMP_FILTER	y	y
CONFIG_SECURITY	y	y
CONFIG_SECURITY_DMESG_RESTRICT	y	y
CONFIG_SECURITY_LANDLOCK	n	n
CONFIG_SECURITY_LOCKDOWN_LSM	y	y
CONFIG_SECURITY_LOCKDOWN_LSM_EARLY	y	y
CONFIG_SECURITY_SELINUX_BOOTPARAM	y	y
CONFIG_SECURITY_SELINUX_DEVELOP	y	y
CONFIG_SECURITY_SELINUX_DISABLE	n	n

Option **CONFIG**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_SECURITY_WRITABLE_HOOKS	N/A	N/A
CONFIG_SECURITY_YAMA	y	y
CONFIG_SHUFFLE_PAGE_ALLOCATOR	y	y
CONFIG_SLAB_FREELIST_HARDENED	y	y
CONFIG_SLAB_FREELIST_RANDOM	y	y
CONFIG_SLUB_DEBUG	y	y
CONFIG_STACKPROTECTOR	y	y
CONFIG_STACKPROTECTOR_STRONG	y	y
CONFIG_STATIC_USERMODEHELPER	n	n
CONFIG_STRICT_DEVMEM	n	n
CONFIG_STRICT_KERNEL_RWX	y	y
CONFIG_STRICT_MODULE_RWX	y	y
CONFIG_SYN_COOKIES	y	y
CONFIG_VMAP_STACK	y	y
CONFIG_WERROR	n	n

Option **CONFIG**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_ZERO_CALL_USED_REGS	n	n

Autoriser l'insertion/le remplacement des méthodes ACPI lors de l'exécution (CONFIG_ACPI_CUSTOM_METHOD)

Amazon Linux désactive cette option, car elle permet aux utilisateurs `root` d'écrire dans une mémoire de noyau arbitraire.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Formats binaires divers (**binfmt_misc**)

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Dans AL2023, cette fonctionnalité est facultative et est construite en tant que module de noyau.

Prise en charge de **BUG()**

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

BUG() si le noyau détecte une corruption de données lors de la vérification de la validité des structures de mémoire du noyau

Certaines parties du noyau Linux vérifient la cohérence interne des structures de données et peuvent générer un élément `BUG()` lorsqu'elles détectent une corruption de données.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

COMPAT_BRK

Lorsque cette option est désactivée (ce qui est la manière dont Amazon Linux configure le noyau), le paramètre `randomize_va_space` `sysctl` est défini par défaut sur 2, ce qui permet également l'aléatorisation des tas en plus de l'aléatorisation des pages VSDO, de la pile et de la base `mmap`.

Cette option existe dans le noyau pour assurer la compatibilité avec certains binaires `libc.so.5` anciens datant de 1996 et avant.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

`COMPAT_VDSO`

Cette option de configuration est pertinente pour `x86-64` et non pour `aarch64`. En définissant ce paramètre sur `n`, le noyau Amazon Linux ne rend pas visible un objet partagé dynamique virtuel (VDSO) 32 bits à une adresse prévisible. La bibliothèque `glibc` la plus récente connue pour être corrompue par la définition de cette option sur `n` est la `glibc` 2.3.3, datant de 2004.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Sécurisation renforcée contrôlée par `CONFIG_DEBUG`

Les options de configuration du noyau Linux contrôlées par `CONFIG_DEBUG` sont généralement destinées aux noyaux conçus pour la résolution des problèmes et pour les situations où les performances ne sont pas une priorité. L'AL2023 permet l'option de `CONFIG_DEBUG_LIST` durcissement.

Désactivation du DMA pour les périphériques PCI dans le stub EFI avant de configurer IOMMU

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

Sécurisation renforcée pour copier la mémoire entre le noyau et l'espace utilisateur

Lorsque le noyau doit copier la mémoire avec l'espace utilisateur comme source ou comme cible, cette option active certaines vérifications qui contribuent à prévenir certaines catégories de problèmes de débordement de mémoire.

L'option `CONFIG_HARDENED_USERCOPY_FALLBACK` existait dans les noyaux 4.16 à 5.15 pour aider les développeurs de noyau à découvrir les entrées manquantes de la liste d'autorisation via un élément `WARN()`. Comme AL2023 est livré avec un noyau 6.1, cette option n'est plus pertinente pour AL2023.

L'`CONFIG_HARDENED_USERCOPY_PAGESPAN`option existait dans les noyaux principalement en tant qu'option de débogage pour les développeurs et ne s'applique plus au noyau 6.1 dans AL2023.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Prise en charge de la mise en veille prolongée

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Cette option doit être activée afin de permettre la mise en veille prolongée de votre instance à la demande et la mise en veille des instances Spot interrompues

Génération de nombres aléatoires

Le noyau AL2023 est configuré pour garantir la disponibilité d'une entropie adéquate pour une utilisation dans EC2.

`CONFIG_INET_DIAG`

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Dans AL2023, cette fonctionnalité est facultative et est construite en tant que module de noyau.

Mise à zéro de toutes les pages du noyau et de la mémoire de l'allocateur de sections lors de l'allocation et de la désallocation

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Ces options sont désactivées dans AL2023 en raison de l'impact possible de l'activation de cette fonctionnalité par défaut en matière de performances. Le comportement `CONFIG_INIT_ON_ALLOC_DEFAULT_ON` peut être activé en ajoutant `init_on_alloc=1` à la ligne de commande du noyau, et le comportement `CONFIG_INIT_ON_FREE_DEFAULT_ON` peut être activé en ajoutant `init_on_free=1`.

Initialise toutes les variables de pile poru qu'elles correspondent à zéro (`CONFIG_INIT_STACK_ALL_ZERO`)

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Cette option nécessite GCC 12 ou version supérieure, tandis qu'AL2023 est fourni avec GCC 11.

Signature des modules du noyau

AL2023 signe et valide les signatures des modules du noyau. L'option `CONFIG_MODULE_SIG_FORCE`, qui oblige les modules à avoir une signature valide, n'est pas

activée afin de préserver la compatibilité pour les utilisateurs qui créent des modules tiers. Pour les utilisateurs qui souhaitent s'assurer que tous les modules du noyau sont signés, le Module de sécurité Linux (LSM) de verrouillage peut être configuré pour l'appliquer.

`kexec`

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Cette option est activée afin que la fonctionnalité `kdump` puisse être utilisée.

Prise en charge de `IOMMU`

AL2023 permet le support de l'IOMMU. L'option `CONFIG_IOMMU_DEFAULT_DMA_STRICT` n'est pas activée par défaut, mais cette fonctionnalité peut être configurée en ajoutant `iommu.passthrough=0 iommu.strict=1` à la ligne de commande du noyau.

`kfence`

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

Prise en charge de l'ancienne version de `pty`

AL2023 utilise l'PTYinterface moderne (`devpts`).

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Module de sécurité Linux (LSM) de verrouillage

AL2023 construit le `lockdown` LSM, qui verrouille automatiquement le noyau lors de l'utilisation du démarrage sécurisé.

L'option `CONFIG_LOCK_DOWN_KERNEL_FORCE_CONFIDENTIALITY` n'est pas activée. Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Lorsque vous n'utilisez pas le démarrage sécurisé, il est possible d'activer le module LSM de verrouillage et de le configurer comme vous le souhaitez.

Empoisonnement des pages

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

De même Mise à zéro de toutes les pages du noyau et de la mémoire de l'allocateur de sections lors de l'allocation et de la désallocation , cela est désactivé dans le noyau AL2023 en raison de l'impact possible sur les performances.

Protecteur de pile

Le noyau AL2023 est construit avec la fonction de protection de pile GCC activée avec l'option. `-fstack-protector-strong`

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

API seccomp BPF

La fonctionnalité de sécurisation renforcée seccomp est utilisée par des logiciels tels que `systemd` et les environnements d'exécution de conteneurs pour renforcer la sécurité des applications de l'espace utilisateur.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Délai d'expiration **`panic()`**

Le noyau AL2023 est configuré avec cette valeur définie sur`0`, ce qui signifie que le noyau ne redémarrera pas après avoir paniqué. Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Cette option est configurable via `sysctl`, `/proc/sys/kernel/panic` et la ligne de commande du noyau.

Modèles de sécurité

AL2023 active SELinux en mode permissif par défaut. Pour plus d'informations, consultez Configuration des modes SELinux pour AL2023.

Les modules Module de sécurité Linux (LSM) de verrouillage et `yama` sont également activés.

`/proc/kcore`

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

Aléatorisation par décalage de la pile du noyau lors d'une entrée syscall

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Cette option peut être activée en configurant `randomize_kstack_offset=on` dans la ligne de commande du noyau.

Vérification du comptage des références (**`CONFIG_REFCOUNT_FULL`**)

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Cette option n'est pas activée actuellement en raison de son impact possible sur les performances.

Connaissance des noyaux SMT par le planificateur (**`CONFIG_SCHED_CORE`**)

Le noyau AL2023 est intégré`CONFIG_SCHED_CORE`, ce qui permet aux applications de l'espace utilisateur de les utiliser. `prctl(PR_SCHED_CORE)` Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Recherche de toute corruption de la pile lors des appels à **`schedule()`** (**`CONFIG_SCHED_STACK_END_CHECK`**)

Le noyau AL2023 est construit avec `CONFIG_SCHED_STACK_END_CHECK` Enabled. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Sécurisation renforcée de l'allocateur de mémoire

Le noyau AL2023 permet de renforcer l'allocateur de mémoire du noyau avec les options`CONFIG_SHUFFLE_PAGE_ALLOCATOR`, et`CONFIG_SLAB_FREELIST_HARDENED`. `CONFIG_SLAB_FREELIST_RANDOM` Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Prise en charge du débogage de SLUB

Le noyau AL2023 est activé `CONFIG_SLUB_DEBUG` car cette option active des fonctionnalités de débogage facultatives pour l'allocateur qui peuvent être activées sur la ligne de commande du noyau. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

CONFIG_STATIC_USERMODEHELPER

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP. Cela est dû au fait que `CONFIG_STATIC_USERMODEHELPER` nécessite une prise en charge spéciale de la part de la distribution, qui n'est actuellement pas présente dans Amazon Linux.

Texte du noyau en lecture seule et rodata (**CONFIG_STRICT_KERNEL_RWX** et **CONFIG_STRICT_MODULE_RWX**)

Le noyau AL2023 est configuré pour marquer le texte et la rodata mémoire du noyau et du module noyau comme étant en lecture seule, et pour marquer la mémoire non textuelle comme non exécutable. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Prise en charge des cookies de synchronisation TCP (**CONFIG_SYN_COOKIES**)

Le noyau AL2023 est conçu avec le support des syncookies TCP. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Pile mappée virtuellement avec pages de protection (**CONFIG_VMAP_STACK**)

Le noyau AL2023 est intégré`CONFIG_VMAP_STACK`, ce qui permet de mapper virtuellement des piles de noyaux avec des pages de protection. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Compilation avec les avertissements du compilateur sous forme d'erreurs (**CONFIG_WERROR**)

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

Enregistrement de la mise à zéro à la sortie de la fonction (**CONFIG_ZERO_CALL_USED_REGS**)

Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

Adresse minimale pour l'allocation d'espace utilisateur

Cette option de sécurisation renforcée contribue à réduire l'impact des bogues de pointeur NULL du noyau. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Options de sécurisation renforcée spécifiques à **`clang`**

Le noyau AL2023 est construit avec GCC plutôt queclang, de sorte que l'option de `CONFIG_CFI_CLANG` renforcement ne peut pas être activée, ce qui la rend également inapplicable`CONFIG_CFI_PERMISSIVE`. Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

Options de sécurisation renforcée du noyau spécifiques à x86-64

Option **`CONFIG`**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
`CONFIG_AMD_IOMMU`	N/A	y
`CONFIG_AMD_IOMMU_V2`	N/A	y
`CONFIG_IA32_EMULATION`	N/A	y
`CONFIG_INTEL_IOMMU`	N/A	y
`CONFIG_INTEL_IOMMU_DEFAULT_ON`	N/A	n
`CONFIG_INTEL_IOMMU_SVM`	N/A	n
`CONFIG_LEGACY_VSYSCALL_NONE`	N/A	n
`CONFIG_MODIFY_LDT_SYSCALL`	N/A	n
`CONFIG_PAGE_TABLE_ISOLATION`	N/A	y
`CONFIG_RANDOMIZE_MEMORY`	N/A	y
`CONFIG_X86_64`	N/A	y

Option **CONFIG**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_X86_MSR	N/A	y
CONFIG_X86_VSYSCALL_EMULATION	N/A	y
CONFIG_X86_X32	N/A	N/A
CONFIG_X86_X32_ABI	N/A	n

Prise en charge de x86-64

La prise en charge de base x86-64 inclut la prise en charge des bits d'extension d'adresse physique (PAE) et de non-exécution (NX). Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Prise en charge d'AMD et Intel IOMMU

Le noyau AL2023 est construit avec le support d'AMD et d'IntelIOMMUs. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

L'option `CONFIG_INTEL_IOMMU_DEFAULT_ON` n'est pas définie, mais elle peut être activée en transmettant `intel_iommu=on` à la ligne de commande du noyau. Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

L'`CONFIG_INTEL_IOMMU_SVM`option n'est actuellement pas activée dans AL2023. Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

Prise en charge de l'espace utilisateur 32 bits

> **Important**
>
> La prise en charge de l'espace utilisateur x86 32 bits est obsolète, et la prise en charge de l'exécution de fichiers binaires d'espace utilisateur 32 bits pourrait être supprimée dans une future version majeure d'Amazon Linux.

> **Note**
>
> Bien qu'AL2023 n'inclue plus de packages 32 bits, le noyau continuera à prendre en charge l'exécution d'un espace utilisateur 32 bits. Pour plus d'informations, consultez Packages x86 (i686) 32 bits.

Pour prendre en charge l'exécution d'applications 32 bits en espace utilisateur, AL2023 n'active pas l'`CONFIG_X86_VSYSCALL_EMULATION`option, mais active les options `CONFIG_IA32_EMULATION`CONFIG_COMPAT, et. `CONFIG_X86_VSYSCALL_EMULATION` Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

L'ABI 32 bits natif x32 pour les processeurs 64 bits n'est pas activé (`CONFIG_X86_X32` et `CONFIG_X86_X32_ABI`). Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Prise en charge du registre spécifique au modèle x86 (MSR)

L'option `CONFIG_X86_MSR` est activée afin de prendre en charge `turbostat`. Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

Syscall **`modify_ldt`**

AL2023 n'autorise pas les programmes utilisateur à modifier la table des descripteurs locaux (LDT) x86 avec l'appel système. `modify_ldt` Cet appel est nécessaire pour exécuter du code 16 bits ou segmenté, et son absence peut corrompre des logiciels tels que `dosemu`, l'exécution de certains programmes sous WINE et de très anciennes bibliothèques de threads. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Suppression du mappage du noyau en mode utilisateur

AL2023 configure le noyau de telle sorte que la majorité des adresses du noyau ne soient pas mappées dans l'espace utilisateur. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Aléatorisation des sections de mémoire du noyau

AL2023 configure le noyau pour randomiser les adresses virtuelles de base des sections de mémoire du noyau. Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Options de sécurisation renforcée du noyau spécifiques à aarch64

Option **CONFIG**	AL2023/6.1/aarch64	AL2023/6.1/x86_64
CONFIG_ARM64_BTI	y	N/A
CONFIG_ARM64_BTI_KERNEL	N/A	N/A
CONFIG_ARM64_PTR_AUTH	y	N/A
CONFIG_ARM64_PTR_AUTH_KERNEL	y	N/A
CONFIG_ARM64_SW_TTBR0_PAN	y	N/A
CONFIG_UNMAP_KERNEL_AT_EL0	y	N/A

Identification de la cible de branche

Le noyau AL2023 permet de prendre en charge l'identification de la cible des branches (`CONFIG_ARM64_BTI`). Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

L'option `CONFIG_ARM64_BTI_KERNEL` n'est pas activée dans AL2023, car elle est construite avec GCC, et la prise en charge de la construction du noyau avec cette option est actuellement désactivée dans le noyau en amont en raison d'un bogue gcc. Bien que cette option soit l'un des paramètres recommandés par le Kernel Self Protection Project (KSPP), AL2023 ne définit pas cette option de configuration selon les recommandations du KSPP.

Authentification par pointeur (`CONFIG_ARM64_PTR_AUTH`)

Le noyau AL2023 prend en charge l'extension Pointer Authentication (qui fait partie des extensions ARMv8.3), qui peut être utilisée pour atténuer les techniques de programmation orientée retour (ROP). La prise en charge du matériel requis pour l'authentification par pointeur sur Graviton a été ajoutée avec Graviton 3.

L'option `CONFIG_ARM64_PTR_AUTH` est activée et prend en charge l'authentification par pointeur pour l'espace utilisateur. Comme l'`CONFIG_ARM64_PTR_AUTH_KERNEL`option est également activée, le noyau AL2023 est en mesure d'utiliser lui-même la protection des adresses de retour.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Émulation de l'accès privilégié Jamais à l'aide de la commutation `TTBR0_EL1`

Cette option empêche le noyau d'accéder directement à la mémoire de l'espace utilisateur. `TTBR0_EL1` n'est défini que temporairement sur une valeur valide par les routines d'accès utilisateur.

Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Annulation du mappage du noyau lors de l'exécution dans l'espace utilisateur

Le noyau AL2023 est configuré pour démapper le noyau lors de l'exécution dans userspace (). `CONFIG_UNMAP_KERNEL_AT_EL0` Cette option est l'un des paramètres recommandés par le Kernel Self Protection Project.

Démarrage sécurisé UEFI sur AL2023

L'AL2023 prend en charge le démarrage sécurisé UEFI à partir de la version 2023.1. Vous devez utiliser AL2023 avec des instances Amazon EC2 qui prennent en charge à la fois UEFI et UEFI Secure Boot. Pour plus d'informations, consultez Lancer une instance dans le guide de l'utilisateur Amazon EC2.

Les instances AL2023 pour lesquelles le démarrage sécurisé UEFI est activé n'acceptent que le code au niveau du noyau, y compris le noyau Linux ainsi que les modules, qui sont signés par. Vous pouvez Amazon donc vous assurer que votre instance n'exécute que des codes au niveau du noyau signés par. AWS

Pour plus d'informations sur les instances Amazon EC2 et le démarrage sécurisé UEFI, consultez la section Démarrage sécurisé UEFI dans le guide de l'utilisateur Amazon EC2.

Prérequis

- Vous devez utiliser une AMI avec AL2023 version 2023.1 ou supérieure.
- Le type d'instance doit prendre en charge UEFI Secure Boot. Pour plus d'informations, consultez Lancer une instance dans le guide de l'utilisateur Amazon EC2.

Activer le démarrage sécurisé UEFI sur AL2023

Les AMI AL2023 standard intègrent un chargeur de démarrage et un noyau signé par nos clés. Vous pouvez activer UEFI Secure Boot en inscrivant des instances existantes ou en créant des AMI avec UEFI Secure Boot préactivé en enregistrant une image à partir d'un instantané. UEFI Secure Boot n'est pas activé par défaut sur les AMI AL2023 standard.

Le mode de démarrage des AMI AL2023 est défini sur `uefi-preferred`, ce qui garantit que les instances lancées avec ces AMI utiliseront le microprogramme UEFI, si le type d'instance prend en charge UEFI. Si le type d'instance ne prend pas en charge UEFI, l'instance est lancée avec le microprogramme BIOS hérité. Lorsqu'une instance est lancée en mode BIOS hérité, UEFI Secure Boot n'est pas appliqué.

Pour plus d'informations sur les modes de démarrage AMI sur les instances Amazon EC2, consultez la section Modes de démarrage du guide de l'utilisateur Amazon EC2.

Rubriques

- Inscription d'une instance existante
- Enregistrement d'une image à partir d'un instantané
- Mises à jour de révocation
- Comment fonctionne le démarrage sécurisé UEFI sur AL2023
- Inscription de vos propres clés

Inscription d'une instance existante

Pour inscrire une instance existante, renseignez les variables de microprogramme UEFI spécifiques avec un jeu de clés qui permettent au microprogramme de vérifier le chargeur de démarrage et à ce dernier de vérifier le noyau lors du prochain démarrage.

1. Amazon Linux fournit un outil pour simplifier le processus d'inscription. Exécutez la commande suivante pour mettre en service l'instance avec le jeu de clés et les certificats nécessaires.

```
sudo amazon-linux-sb enroll
```

2. Exécutez la commande suivante pour redémarrer l'instance. Une fois l'instance redémarrée, UEFI Secure Boot est activé.

```
sudo reboot
```

> **Note**
>
> Actuellement, les AMI Amazon Linux ne prennent pas en charge Nitro Trusted Platform Module (NitroTPM). Si vous avez besoin de NitroTPM en plus d'UEFI Secure Boot, utilisez les informations de la section suivante.

Enregistrement d'une image à partir d'un instantané

Lorsque vous enregistrez une AMI à partir d'un instantané d'un volume racine Amazon EBS à l'aide de l'API Amazon EC2 `register-image`, vous pouvez configurer l'AMI avec un blob binaire contenant l'état du magasin de variables UEFI. En fournissant l'`UefiData` AL2023, vous activez UEFI Secure Boot et vous n'avez pas besoin de suivre les étapes de la section précédente.

Pour plus d'informations sur la création et l'utilisation d'un blob binaire, consultez l'option B : créer un blob binaire contenant un magasin de variables prérempli dans le guide de l'utilisateur Amazon EC2.

AL2023 fournit un blob binaire prédéfini qui peut être utilisé directement sur les instances Amazon EC2. Le blob binaire se trouve dans `/usr/share/amazon-linux-sb-keys/uefi.vars` sur une instance en cours d'exécution. Ce blob est fourni par le package RPM `amazon-linux-sb-keys` qui est installé par défaut sur les AMI AL2023 à partir de la version 2023.1.

> **Note**
>
> Pour vous assurer que vous utilisez la dernière version des clés et des révocations, utilisez le blob de la même version d'AL2023 que celle que vous avez utilisée pour créer l'AMI.

Lors de l'enregistrement d'une image, nous vous recommandons d'utiliser le paramètre `BootMode` de l'API `RegisterImage` défini sur `uefi`. Cela vous permet d'activer NitroTPM en définissant le

paramètre `TpmSupport` sur `v2.0`. En outre, définir `BootMode` sur `uefi` garantit que UEFI Secure Boot est activé et ne peut pas être désactivé par accident lors du passage à un type d'instance qui ne prend pas en charge UEFI.

Pour plus d'informations sur NitroTPM, consultez NitroTPM dans le guide de l'utilisateur Amazon EC2.

Mises à jour de révocation

Il est parfois nécessaire qu'Amazon Linux distribue une nouvelle version du chargeur de démarrage `grub2` ou du noyau Linux signée avec des clés mises à jour. Dans ce cas, il peut être nécessaire de révoquer l'ancienne clé pour éviter que des bogues exploitables provenant des versions précédentes du chargeur de démarrage ne contournent le processus de vérification d'UEFI Secure Boot.

Les mises à jour du package vers les packages `grub2` ou `kernel` mettent toujours à jour automatiquement la liste des révocations dans le magasin de variables UEFI de l'instance en cours d'exécution. Cela signifie que quand UEFI Secure Boot est activé, vous ne pouvez plus exécuter l'ancienne version d'un package après avoir installé une mise à jour de sécurité pour le package.

Comment fonctionne le démarrage sécurisé UEFI sur AL2023

Contrairement aux autres distributions Linux, Amazon Linux ne fournit pas de composant supplémentaire, appelé shim, servant de premier chargeur de démarrage. Le shim est généralement signé avec des clés Microsoft. Par exemple, sur les distributions Linux avec le shim, celui-ci charge le chargeur de démarrage `grub2` qui utilise le propre code du shim pour vérifier le noyau Linux. En outre, le shim conserve son propre jeu de clés et de révocations dans la base de données MOK (Machine Owner Key) située dans le magasin de variables UEFI et contrôlée par l'outil `mokutil`.

Amazon Linux ne fournit pas de shim. Étant donné que le propriétaire de l'AMI contrôle les variables UEFI, cette étape intermédiaire n'est pas nécessaire et aurait une incidence négative sur les temps de lancement et de démarrage. Nous avons également choisi de ne pas inclure la confiance envers les clés des fournisseurs par défaut, afin de réduire le risque d'exécution de fichiers binaires indésirables. Comme toujours, les clients peuvent inclure des fichiers binaires s'ils le souhaitent.

Avec Amazon Linux, UEFI charge et vérifie directement notre chargeur de démarrage `grub2`. Le chargeur de démarrage `grub2` a été modifié pour utiliser UEFI afin de vérifier le noyau Linux après son chargement. Ainsi, le noyau Linux est vérifié à l'aide des mêmes certificats stockés dans la variable db UEFI normale (base de données de clés autorisées) et testé par rapport à la même variable `dbx` (base de données de révocations) que le chargeur de démarrage et les autres fichiers

binaires UEFI. Étant donné que nous fournissons nos propres clés PK et KEK, qui contrôlent l'accès à la base de données db et à la base de données dbx, nous pouvons distribuer des mises à jour et des révocations signées selon les besoins sans passer par un intermédiaire comme le shim.

Pour plus d'informations sur le démarrage sécurisé UEFI, consultez Comment fonctionne le démarrage sécurisé UEFI dans le guide de l'utilisateur Amazon EC2.

Inscription de vos propres clés

Comme indiqué dans la section précédente, Amazon Linux n'a pas besoin d'un `shim` pour UEFI Secure Boot sur Amazon EC2. Lorsque vous lisez la documentation d'autres distributions Linux, vous trouverez peut-être de la documentation sur la gestion de la base de données MOK (Machine Owner Key) à l'aide de `mokutil`, qui n'est pas présent sur AL2023. Les environnements de `shim` et de MOK contournent certaines limites relatives à l'inscription des clés dans le microprogramme UEFI qui ne s'appliquent pas à la manière dont Amazon EC2 implémente UEFI Secure Boot. Amazon EC2 propose des mécanismes permettant de manipuler facilement et directement les clés dans le magasin de variables UEFI.

Si vous souhaitez inscrire vos propres clés, vous pouvez soit manipuler le magasin de variables au sein d'une instance existante (consultez Ajout de clés au magasin de variables depuis l'instance), soit construire un blob binaire prérempli (consultez Création d'un blob binaire contenant un magasin de variables prérempli).

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.