Livre Blanc AWS

La conception de la sécurité du Système AWS Nitro

La conception de la sécurité du Système AWS Nitro: Livre Blanc AWS

Table of Contents

La conception de la sécurité du système AWS Nitro

Date de publication: 18 novembre 2022 (Révisions du document)

Résumé

Amazon Elastic Compute Cloud (Amazon EC2) est un service Web qui fournit une capacité de calcul sécurisée et redimensionnable dans le cloud. Destiné aux développeurs, il facilite l'accès au cloud computing à l'échelle du Web. Le Système AWS Nitro est la plate-forme sous-jacente de toutes les instances EC2 modernes. Ce livre blanc fournit une description détaillée de la conception de la sécurité du Système Nitro afin de vous aider à évaluer EC2 pour vos applications sensibles.

Introduction

Chaque jour, des clients du monde entier confient à Amazon Web Services (AWS) leurs applications les plus sensibles. Chez AWS, assurer la sécurité et la confidentialité des applications de nos clients, tout en les aidant à répondre à leurs exigences en matière de sécurité, de confidentialité et de protection des données, est notre priorité absolue. Nous avons investi dans des pratiques opérationnelles rigoureuses et des technologies de sécurité qui répondent et dépassent même les besoins de sécurité des données de nos clients les plus exigeants.

Le développement du Système AWS Nitro a été un processus de plusieurs années visant à réinventer l'infrastructure de virtualisation d'Amazon EC2. Depuis le lancement de la version bêta d'Amazon EC2 en 2006, nous n'avons cessé d'affiner, d'optimiser et d'innover dans tous les aspects du service afin de répondre aux besoins de nos clients. Avec le système AWS Nitro, nous nous sommes efforcés de repenser radicalement l'architecture de virtualisation afin de fournir la sécurité, l'isolation, les performances, les coûts et le rythme d'innovation dont nos clients ont besoin.

La sécurité a été un principe fondamental de ce développement depuis l’origine, et nous avons continué à investir dans le cadre de l'amélioration continue afin de toujours augmenter la sécurité et la protection des données pour nos clients. Le système AWS Nitro est une combinaison de serveurs physiques, de processeurs, de composants de gestion et de microprogrammes (firmwares) spécialisés qui fournissent la plate-forme sous-jacente à toutes les instances Amazon EC2 lancées depuis le début de l’année 2018. Ensemble, ces composants permettent aux clients Amazon EC2 d'innover plus rapidement, de renforcer la sécurité et d'améliorer les performances.

Trois composants clés du Système Nitro permettent d'atteindre ces objectifs :

- Les Cartes Nitro — dispositifs matériels conçus par AWS qui assurent le contrôle global du système et la virtualisation des entrées/sorties (E/S), indépendamment de la carte mère, de ses processeurs et de sa mémoire.
- La Puce de Sécurité Nitro — permet un démarrage (boot) sécurisé pour l'ensemble du système en s'appuyant sur des éléments matériels de confiance, permet de proposer des instances « bare metal », et fournit une défense en profondeur qui protège le serveur contre toute modification non autorisée du microprogramme (firmware) du système.
- L'Hyperviseur Nitro — hyperviseur délibérément minimal et semblable à un microprogramme (firmware), conçu pour fournir une isolation des ressources solide et des performances quasiment identiques à celles d'un serveur « bare metal ».

> **Note**
>
> Ces composants sont complémentaires mais n'ont pas besoin d'être utilisés ensemble.

Ce document fournit une introduction de haut niveau à la virtualisation et au changement architectural fondamental introduit par le Système Nitro. Il décrit chacun des trois composants clés du Système Nitro et montre comment ces composants fonctionnent ensemble, en expliquant ce qui se passe lorsqu'un nouveau volume Amazon Elastic Block Store (Amazon EBS) est ajouté à une instance EC2 en cours d'exécution. Le livre blanc explique comment le Système Nitro, de par sa conception, élimine la possibilité d'accès administrateur à un serveur EC2, la conception passive des communications du Système Nitro et le processus de gestion des modifications du Système Nitro. Enfin, le livre blanc présente les aspects importants de la conception du système EC2 permettant d'atténuer les risques liés aux potentielles attaques par canaux auxilliaires qui pourraient survenir dans les environnements informatiques.

La virtualisation traditionnelle

En synthèse, la virtualisation permet à un seul ordinateur physique d'exécuter plusieurs systèmes d'exploitation à la fois. Un système de virtualisation (« hôte ») met en œuvre des fonctions de traduction, d'émulation et de restriction qui lui permettent de fournir à un ou plusieurs systèmes d'exploitation virtualisés (« invités ») des représentations virtuelles des capacités matérielles sous-jacentes (« machines virtuelles » ou « VM »). C'est ce que l'on appelle un hôte de virtualisation. L'un des principaux avantages de la virtualisation réside dans la possibilité d'utiliser efficacement un seul serveur physique puissant en répartissant ses ressources entre plusieurs machines virtuelles auxquelles est allouée une quantité de ressources optimale pour les tâches qui lui sont assignées.

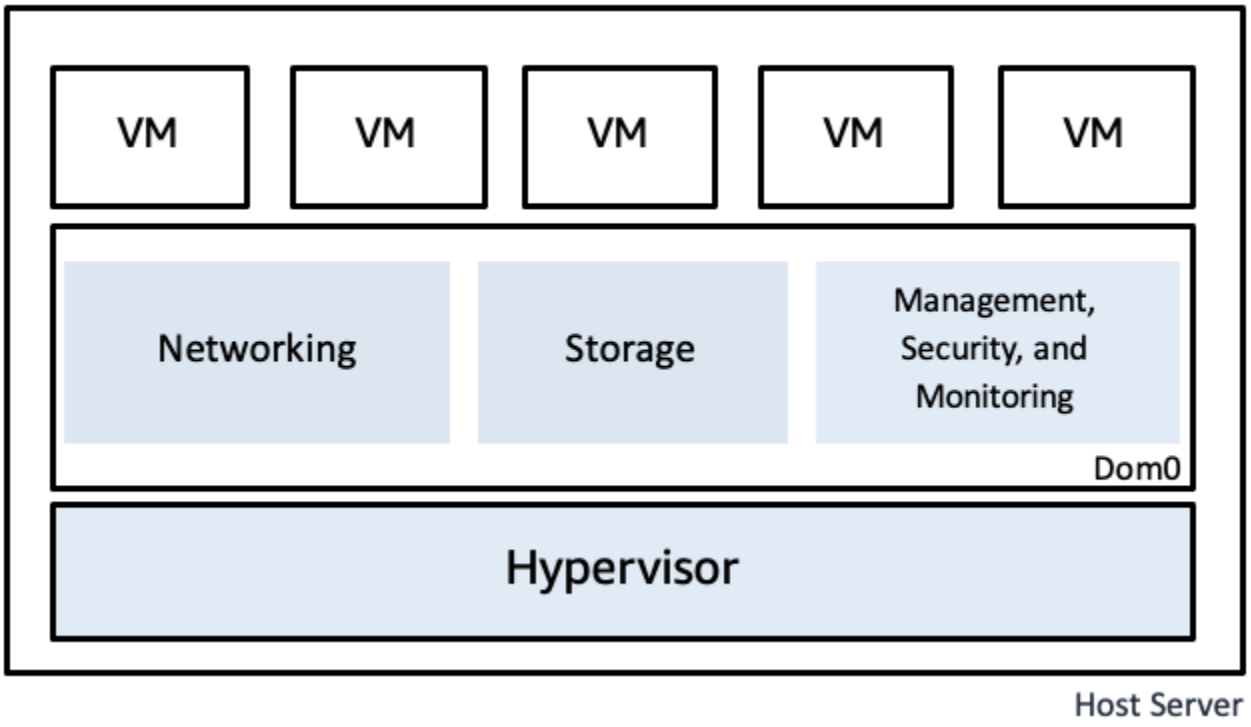
> **Note**
>
> La discussion sur la virtualisation présentée dans cette section fournit une introduction générale de haut niveau et n'aborde pas des sujets tels que la paravirtualisation, dans laquelle le logiciel invité doit être modifié pour fonctionner dans un environnement virtualisé. Pour une présentation plus détaillée des technologies de virtualisation, consultez cette présentation sur les technologies de virtualisation par Anthony Liguori, vice-président et ingénieur émérite chez AWS.

Le composant principal responsable de la gestion du cycle de vie et du fonctionnement des machines virtuelles (VM) invitées dans un système de virtualisation est appelé moniteur de machine virtuelle (VMM), ou hyperviseur. Pour la majorité des opérations qu'il effectue, un invité exécute des instructions de manière native sur le processeur physique du système, sans aucune intervention de la VMM. Par exemple, lorsqu'un client cherche à calculer la somme ou le produit de deux valeurs, il peut communiquer directement avec le processeur du système pour émettre les instructions de code machine requises.

Il existe toutefois certaines classes d'instructions sensibles ou privilégiées, telles que la lecture ou l'écriture à partir de registres de contrôle du processeur, qu'un invité ne doit pas être autorisé à exécuter directement sur le matériel du processeur afin de maintenir la stabilité et l'isolation du système dans son ensemble. Lorsqu'un client essaie de transmettre l'une de ces instructions au processeur, au lieu de l'exécuter, l'instruction est redirigée vers le VMM, qui émule un résultat autorisé pour l'instruction, puis rend le contrôle à l'invité, comme si l'instruction avait été exécutée directement sur le processeur.

Le VMM lui-même est un logiciel relativement simple. Cependant, un hôte de virtualisation a besoin de plus de fonctionnalités pour permettre aux clients d'accéder à des périphériques tels que des interfaces réseau, des disques de stockage et des périphériques d'entrée. Pour fournir ces fonctionnalités, les hôtes s'appuient sur des composants logiciels supplémentaires appelés modèles de périphériques (« device models »). Les modèles de périphériques communiquent avec le matériel physique d'E/S partagé du système et émulent l'état et le comportement d'une ou de plusieurs interfaces de périphériques virtuels exposées aux machines virtuelles clientes.

Les hyperviseurs utilisent généralement un système d'exploitation standard pour s'interfacer avec divers matériels du système et exécuter des modèles de périphériques ainsi que d'autres logiciels de gestion pour le système de virtualisation. Ce système d'exploitation est généralement implémenté sous la forme d'une machine virtuelle à privilèges spéciaux appelée dom0 dans le projet Xen, et partition root/parent du système par Hyper-V . Dans les instances EC2 de première génération, cela prenait la forme d'une machine virtuelle Amazon Linux spéciale s'exécutant sous le nom de domaine 0, ou dom0 dans la terminologie Xen.

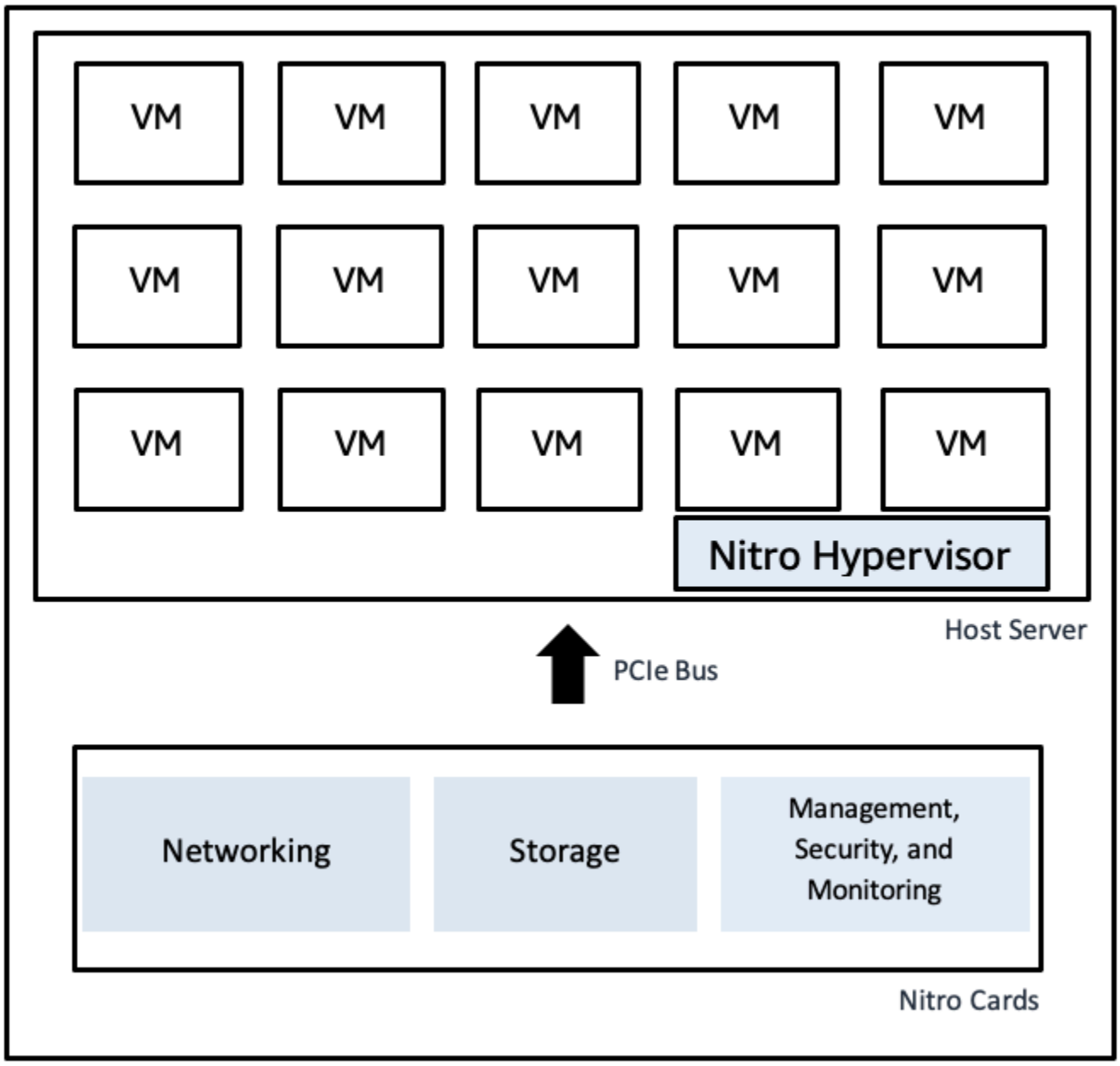


Architecture traditionnelle de virtualisation

L'évolution vers le Système Nitro

Le Système Nitro est le produit d'un parcours de plusieurs années visant à réinventer la technologie de virtualisation pour l'infrastructure du Cloud AWS. Au cours de ce parcours, chaque composant de la virtualisation a été réimplémenté et remplacé. Alors que les clients ont constaté une amélioration des coûts, des performances et de la sécurité grâce aux instances EC2 mises en services les premières années, les instances basées sur le Système Nitro complet, dans lequel chaque composant a été remplacé, sont nettement différentes des types d'instances précédents. Le Système Nitro offre une sécurité, une confidentialité et des performances améliorées aux clients d'Amazon EC2, et permet de fournir des technologies innovantes à un rythme rapide.

L'introduction du Système Nitro a consisté en une décomposition progressive des composants logiciels exécutés dans Dom0 sur un processeur standard, en unités de coprocesseur indépendantes spécialement conçues à cet effet. Ce qui a commencé comme un système de virtualisation monolithique a été progressivement transformé en une architecture de microservices spécialisés. En commençant par le type d'instance C5 introduit en 2017, le Système Nitro a totalement éliminé le besoin de Dom0 sur une instance EC2. Au lieu de cela, un hyperviseur minimal développé sur mesure et basé sur KVM fournit un VMM allégé, tout en déchargeant d'autres fonctions, telles que celles précédemment exécutées par les modèles de périphérique de Dom0, dans un ensemble de Cartes Nitro.



Architecture de virtualisation du Système Nitro

Les composants du Système Nitro

Comme indiqué précédemment, le Système Nitro est constitué de trois composants principaux :

- Les Cartes Nitro
- La Puce de Sécurité Nitro
- L'Hyperviseur Nitro

Les Cartes Nitro

Un serveur EC2 moderne est composé d'une carte système principale et d'une ou de plusieurs Cartes Nitro. La carte principale, ou carte mère, contient les processeurs hôtes (processeurs Intel, AMD ou Graviton) et la mémoire. Les Cartes Nitro sont des composants matériels dédiés dotés de puissantes capacités de traitement 64 bits et de circuits intégrés spécifiques aux applications (« ASIC ») qui fonctionnent indépendamment de la carte mère qui, elle, exécute les environnements informatiques du client, y compris les opérations de traitement du code et des données.

Les Cartes Nitro mettent en oeuvre toutes les interfaces de contrôle externes utilisées par le service EC2 pour provisionner et gérer la puissance de calcul, la mémoire et le stockage. Ils fournissent également toutes les interfaces d'E/S, telles que celles nécessaires pour fournir le réseau virtuel (SDN – Software Defined Network), le stockage Amazon EBS et le stockage dédié à l’instance (instance storage). Cela signifie que tous les composants qui interagissent avec le monde extérieur du service EC2, au-delà de la carte mère, qu'ils soient entrants ou sortants, s'exécutent sur des composants informatiques autonomes physiquement séparés de la carte mère sur laquelle s'exécutent les environnements des clients.

Le Système Nitro est conçu pour créer une forte isolation logique entre les composants de l’hôte et les Cartes Nitro ; ceci bénéficie de l’isolation physique décrite précédemment, qui fournit une délimitation claire et fiable entre ces composants. Tout en étant ainsi isolées logiquement et séparées physiquement, les Cartes Nitro se trouvent généralement dans le même boîtier physique que la carte mère du système hôte et partagent son alimentation ainsi que son interface PCIe.

> **Note**
>
> Dans le cas des instances EC2 mac1.metal et mac2.metal, un Contrôleur Nitro est colocalisé avec un Mac Mini, dans un boîtier physique commun, et les deux sont connectés via Thunderbolt. Reportez-vous à la section Amazon EC2 Mac Instances pour plus de détails.

Les principaux composants des Cartes Nitro sont constitués d'un système embarqué sur un circuit intégré (System on a Chip - SoC) conçu par AWS, qui exécute un micrologiciel (firmware) spécialisé. AWS a soigneusement piloté le processus de conception et de fabrication du matériel et du microprogramme de ces cartes. Le matériel est entièrement conçu par Annapurna Labs, l'équipe responsable de la conception interne des composants électroniques d'AWS. Le micrologiciel de ces cartes est développé et maintenu par des équipes d'ingénierie d'AWS dédiées.

> **Note**
>
> Annapurna Labs a été rachetée par Amazon en 2015, suite à un partenariat réussi lors des phases initiales de développement des principales technologies du système AWS Nitro. Annapurna Labs est responsable non seulement de la fabrication du matériel du système AWS Nitro, mais également des processeurs Graviton spécialement conçus et réalisés pour AWS et basés sur la technologie ARM, des puces d'accélération matérielle AWS Trainium et AWS Inferentia pour l'entrainement et l'inférence en apprentissage automatique (machine learning), du SSD AWS Nitro SSD, et d'Aqua (accélérateur de requêtes avancé) pour Amazon Redshift.

Le microprogramme de contrôle des Cartes Nitro peut être mis à jour en direct à l'aide de progiciels signés cryptographiquement. Les Cartes Nitro peuvent être mises à jour indépendamment des autres composants du Système Nitro, y compris les unes des autres et de tout composant susceptible d'être mis à jour sur la carte mère, de façon à déployer de nouvelles fonctionnalités et des mises à jour de sécurité. La mise à jour des Cartes Nitro a un impact quasi-imperceptible sur les applications des clients et ne nécessite aucun assouplissement des contrôles de sécurité du Système Nitro.

Les Cartes Nitro sont connectées physiquement à la carte mère du système et à ses processeurs via PCIe, mais elles sont isolées logiquement de la carte mère qui exécute les environnements des clients. Un Système Nitro peut contenir une ou plusieurs Cartes Nitro ; s'il en existe plusieurs, elles sont connectées via un réseau interne au sein d'un boîtier physique. Ce réseau fournit un canal de communication privé entre les Cartes Nitro, indépendant de la carte mère, ainsi qu'une connexion

privée au contrôleur de gestion de la carte mère (Baseboard Management Controller, BMC), s'il en existe un dans le serveur.

Le Contrôleur Nitro

La carte Nitro principale est appelée Contrôleur Nitro. Le Contrôleur Nitro constitue la base de confiance (root of trust) matérielle de l'ensemble du système. Il a pour fonction de gérer tous les autres composants du système, y compris le microprogramme chargé sur les autres composants. Le microprogramme du système dans son ensemble est stocké sur une mémoire SSD chiffrée, directement connectée au Contrôleur Nitro. La clé de chiffrement de la mémoire SSD est protégée par la combinaison d'un module TPM (Trusted Platform Module) et des fonctionnalités de démarrage sécurisé du SoC. Cette section décrit comment est conçu le démarrage sécurisé du Contrôleur Nitro ainsi que le rôle de ce dernier comme interface sécurisée entre un serveur et le réseau.

> **Note**
>
> Dans le cas des déploiements d'AWS Outpost, une clé de sécurité Nitro est également utilisée avec un module TPM et les fonctionnalités de démarrage sécurisé du SoC pour protéger la clé de chiffrement de la mémoire SSD, qui est directement connectée au Contrôleur Nitro.

Le démarrage sécurisé du Contrôleur Nitro

Le processus de démarrage sécurisé du SoC du Contrôleur Nitro commence par sa ROM (mémoire morte) de démarrage, et se poursuit avec la chaîne de confiance en surveillant et en vérifiant les premiers stades du microprogramme stocké dans une mémoire flash connectée au Contrôleur Nitro. Au fur et à mesure que l'initialisation du système progresse, un TPM (Trusted Platform Module) est utilisé pour enregistrer les mesures initiales du code de démarrage, puis pour étendre les mesures à d'autres microprogrammes du système. Les clés cryptographiques intégrées au module TPM infalsifiable sont utilisées pour signer numériquement l'ensemble complet des mesures de référence du système. Ce fichier signé numériquement est ensuite comparé à toutes les mesures système suivantes à chaque redémarrage afin de détecter toute modification inattendue.

Si aucune modification n'est détectée, des clés de déchiffrement supplémentaires, elles-mêmes chiffrées par des clés verrouillées dans le module TPM, sont utilisées pour déchiffrer des données supplémentaires dans le système afin de permettre au processus de démarrage de se poursuivre. Si

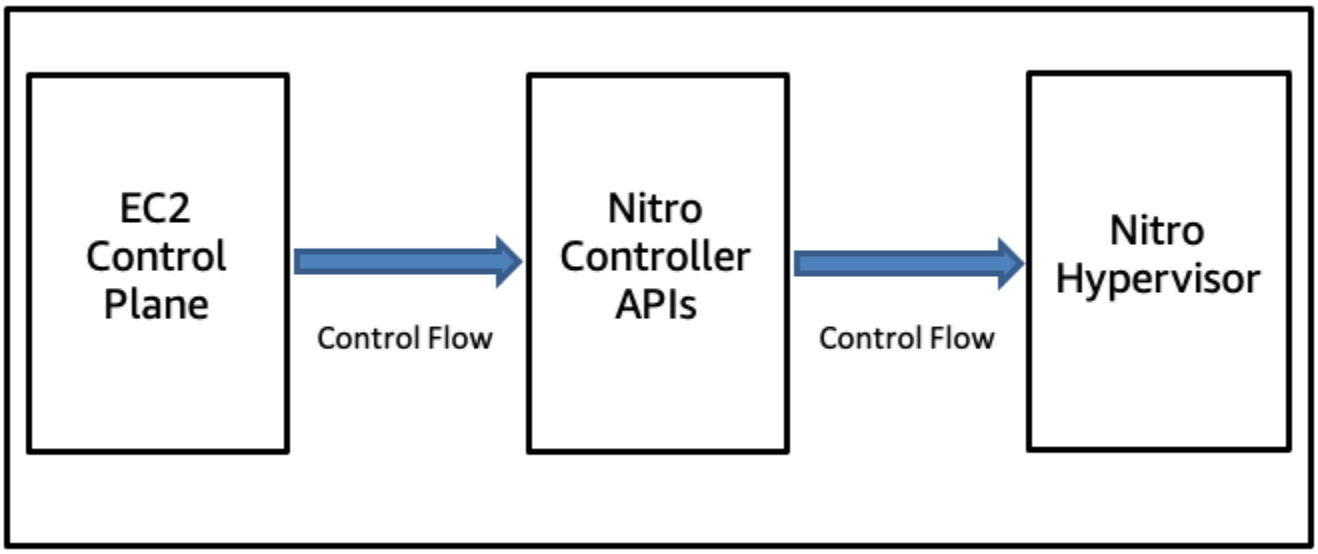
des modifications sont détectées, les données supplémentaires ne sont pas déchiffrées et le système est immédiatement mis hors service et n'hébergera ainsi pas les environnements des clients.

Les étapes précédentes détaillent le processus par lequel le Contrôleur Nitro vérifie l'intégrité et la validité du logiciel système au démarrage. Pour que la procédure de démarrage sécurisé soit réellement sécurisée, chaque étape du code de démarrage du SoC doit non seulement être valide et non modifiée, mais aussi fonctionnellement correcte telle qu'elle est implémentée. Cela est particulièrement le cas du code ROM statique qui fait partie de la fabrication physique du composant. À cette fin, AWS a, lors de la conception intégré des méthodes de preuve formelles pour vérifier les propriétés de sécurité de la mémoire du code de démarrage initial.

Le Contrôleur Nitro comme interface entre les serveurs EC2 et le réseau

Le Contrôleur Nitro est la passerelle exclusive entre le serveur physique et les plans de contrôle pour EC2, Amazon EBS et Amazon Virtual Private Cloud (Amazon VPC). Bien que logiquement distincts et composés de plusieurs microservices, ces trois plans de contrôle sont désignés ci-après par plan de contrôle EC2..

> **Note**
>
> Au sein d'AWS, un modèle de conception courant consiste à diviser un système en services chargés de traiter les commandes informatiques générées par les environnements des clients (le plan de données – « data plane ») et services chargés de gérer et de distribuer la configuration des ressources des clients, par exemple pour créer, supprimer ou en modifier des ressources (le plan de contrôle – « control plane »). Amazon EC2 est un exemple de service qui inclut un plan de données et un plan de contrôle. Le plan de données se compose de serveurs physiques EC2 sur lesquels s'exécutent les instances virtuelles EC2 des clients. Le plan de contrôle comprend un certain nombre de services chargés de communiquer avec le plan de données et d'exécuter des fonctions telles que le relais des commandes pour lancer ou terminer une instance ou l'ingestion des données de télémétrie.



Architecture de contrôle du Système Nitro

Le Contrôleur Nitro présente au réseau dédié au plan de contrôle EC2 un ensemble d'API réseau avec une authentification et un chiffrement robustes, permettant la gestion du système. Chaque action d'interface API est enregistrée et toutes les tentatives d'appel d'une API sont authentifiées et autorisées par chiffrement à l'aide d'un système fin de contrôle d'accès. Chaque composant du plan de contrôle est autorisé à réaliser uniquement les opérations nécessaires à la réalisation de sa mission. Nous avons utilisé des méthodes formelles pour démontrer que l'API analysant les messages de contrôle du Contrôleur Nitro est exempte de faille de sécurité mémoire, quelle que soit le fichier de configuration et quelle que soit l’entrée réseau.

Cartes Nitro pour les entrées/sorties (E/S)

Outre le Contrôleur Nitro, certains systèmes utilisent des Cartes Nitro spécialisées supplémentaires pour exécuter certaines fonctions spécifiques. Ces Cartes Nitro subordonnées partagent la même architecture de SoC et les mêmes microprogrammes de base que le Contrôleur Nitro. Elles sont conçues avec du matériel supplémentaire et des microprogrammes spécialisés pour leurs fonctions spécifiques. Il s'agit, par exemple, de la Carte Nitro pour VPC, de la Carte Nitro pour EBS et de la Carte Nitro pour le stockage NVMe Local.

Ces cartes mettent en œuvre le chiffrement des données pour la mise en réseau et le stockage à l'aide de moteurs matériels dotés d'un stockage sécurisé des clés intégré au SoC. Ces moteurs matériels permettent de chiffrer à la fois le stockage NVMe local et les volumes EBS distants sans impact mesurable sur leurs performances. Les trois dernières versions de la Carte Nitro pour VPC,

y compris celles utilisées sur tous les types d'instances récemment mises en service, chiffrent de manière transparente et sans impact sur les performances tout le trafic VPC vers d'autres instances EC2 exécutées sur des hôtes également équipés de Cartes Nitro compatibles avec le chiffrement.

> **Note**
>
> AWS fournit une connectivité sécurisée et privée entre les instances EC2 de tous types. En outre, certains types d'instances utilisent les fonctionnalités de déchargement du matériel du Système Nitro sous-jacent pour chiffrer et anonymiser automatiquement et de manière transparente le trafic en transit entre les instances, à l'aide du protocole AES-256-GCM. Cela n'a aucun impact sur les performances du réseau. Pour que ce chiffrement supplémentaire du trafic en transit entre les instances soit mis en œuvre, les instances doivent appartenir à des types d'instance pris en charge, être situées dans la même région et dans le même VPC ou dans des VPC appairés. Le trafic ne doit pas passer par un périphérique ou un service réseau virtuel tel qu'un équilibreur de charge ou une passerelle de transit. Pour plus d'informations et la liste des types d'instances pris en charge, consultez la section Chiffrement en transit.

Les clés de chiffrement utilisées pour EBS, le stockage d'instance local et les réseaux VPC ne sont présentes en clair que dans la mémoire volatile protégée des Cartes Nitro ; elles sont inaccessibles à la fois aux opérateurs AWS et à tout code client exécuté sur les processeurs du système hôte. Les cartes Nitro fournissent des interfaces de programmation (API) matérielles via la connexion PCIe au processeur du serveur physique : NVMe pour le stockage par blocs (EBS et stockage d'instances), Elastic Network Adapter (ENA) pour la mise en réseau, port série pour un accès « out-of-band » aux consoles du système d'exploitation pour la journalisation et le débogage, etc.

> **Note**
>
> EC2 permet aux clients d'accéder à la sortie de la console de l'instance à des fins de dépannage. Le Système Nitro permet également aux clients de se connecter à une session de la console série pour résoudre de manière interactive les problèmes de démarrage, de configuration réseau et autres. Dans ce contexte, le terme « out-of-band » désigne la capacité des clients à obtenir des informations ou à interagir avec leurs instances via un canal distinct de l'instance elle-même ou de sa connexion réseau.

Lorsqu'un système est configuré pour utiliser l'Hyperviseur Nitro, chaque fonction PCIe fournie par une Carte Nitro est subdivisée en fonctions virtuelles à l'aide de la technologie de virtualisation des entrées/sorties à racine unique (SR-IOV). Cela facilite l'attribution d'interfaces matérielles directement aux machines virtuelles. Les données des clients (le contenu que les clients nous transfèrent à des fins de traitement, de stockage ou d'hébergement) sont transférées directement entre les instances et ces périphériques d'E/S virtualisés fournis par les Cartes Nitro. Cela permet de réduire le nombre de composants logiciels et matériels utilisés dans les E/S, ce qui se traduit par une baisse des coûts, de meilleures performances et une sécurité accrue.

La Puce de Sécurité Nitro

Le Contrôleur Nitro et les autres Cartes Nitro fonctionnent ensemble comme un seul domaine dans un Système Nitro, tandis que la carte mère, dotée de processeurs Intel, AMD ou Graviton, sur laquelle s'exécutent les environnement des clients, constitue un second domaine distinct. Bien que le Contrôleur Nitro et son processus de démarrage sécurisé constituent les éléments matériels de base de la confiance d’un Système Nitro, un composant supplémentaire est utilisé pour étendre cette confiance à la carte mère du système. La Puce de Sécurité Nitro est le lien entre ces deux domaines et étend le contrôle du Contrôleur Nitro à la carte mère et en fait un composant subordonné du système couvert par la chaîne de confiance du Contrôleur Nitro. Les sections suivantes expliquent comment le Contrôleur Nitro et la Puce de Sécurité Nitro fonctionnent ensemble pour atteindre cet objectif.

La protection du matériel système par la Puce de Sécurité Nitro

La Puce de Sécurité Nitro est un composant intégré à la carte mère du serveur. Au moment de l'exécution, elle intercepte et modère toutes les opérations vers les composants de stockage non volatil locaux et les interfaces de bus de gestion système à faible vitesse (telles que l'interface périphérique série (SPI) et l'I^2C), c’est-à-dire vers tous les microprogrammes.

La Puce de Sécurité Nitro est située entre le BMC et le processeur principal, sur sa connexion PCI haut débit, ce qui constitue un pare-feu logique pour protéger cette interface sur les systèmes de production. La Puce de Sécurité Nitro est contrôlée par le Contrôleur Nitro. C’est par l’intermédiaire de son contrôle sur la Puce de Sécurité Nitro que le Contrôleur Nitro gère et valide les mises à jour des micrologiciels et la programmation d'autres composants non volatils sur la carte mère ou les Cartes Nitro.

Une pratique courante consiste à s'appuyer sur un hyperviseur pour protéger ces actifs matériels, mais en l'absence d'hyperviseur, par exemple lorsque EC2 est utilisé en mode « bare metal », un

autre mécanisme est nécessaire pour s'assurer que les utilisateurs ne peuvent pas manipuler le microprogramme du système. La Puce de Sécurité Nitro assure la fonction de sécurité essentielle qui garantit que les processeurs principaux du système ne peuvent pas mettre à jour le micrologiciel en mode « bare metal ». De plus, lorsque le Système Nitro fonctionne avec l'Hyperviseur Nitro, la Puce de Sécurité Nitro fournit également une défense en profondeur qui s'ajoute à la protection des composants matériels du système assurée par l'hyperviseur.

La Puce de Sécurité Nitro au démarrage ou à la réinitialisation du système

La Puce de Sécurité Nitro remplit également une autre fonction de sécurité critique lors du démarrage ou de la réinitialisation du système. Elle contrôle les broches physiques de réinitialisation de la carte mère du système, y compris ses processeurs et le contrôleur de gestion de la carte mère (BMC), le cas échéant. Cela permet au Contrôleur Nitro d'accomplir son propre processus de démarrage sécurisé, puis de vérifier l'intégrité du système d'entrée/sortie de base (BIOS), du BMC et de tous les autres microprogrammes avant de commander à la Puce de Sécurité Nitro de libérer les processeurs principaux et le BMC. Ce n'est qu'ensuite que les processeurs et le BMC peuvent commencer leur processus de démarrage à l'aide du BIOS et du micrologiciel qui viennent d'être validés par le Contrôleur Nitro.

L'Hyperviseur Nitro

L'Hyperviseur Nitro est un composant conçu intentionnellement pour être réduit et strictement limité aux fonctionnalités nécessaires pour exécuter les fonctions qui lui sont assignées, et rien de plus. L'Hyperviseur Nitro est conçu pour recevoir les commandes de gestion des machines virtuelles (démarrage, arrêt, etc.) envoyées par le Contrôleur Nitro, pour partitionner la mémoire et les ressources du processeur en utilisant les fonctionnalités de virtualisation matérielle du processeur, et pour attribuer des fonctions virtuelles SR-IOV fournies par les interfaces matérielles Nitro (stockage par blocs NVMe pour le stockage EBS et le stockage d'instance, Elastic Network Adapter (ENA) pour le réseau, etc.) à la machine virtuelle appropriée, via PCIe.

> **Note**
>
> L'architecture Nitro est unique du fait qu'elle ne nécessite pas d'hyperviseur pour fournir des composants d'infrastructure définis par logiciel (« software defined »). Néanmoins, dans la plupart des scénarios clients, la virtualisation est utile car elle permet de subdiviser de très grands serveurs pour une utilisation simultanée en plusieurs instances (VM) et présente d'autres avantages tels qu'un provisionnement plus rapide. Les hyperviseurs sont

nécessaires dans les configurations virtualisées pour fournir l'isolation, la planification et la gestion des invités et du système. L'Hyperviseur Nitro joue donc un rôle essentiel dans la sécurisation d'un serveur EC2 basé sur Nitro dans ces scénarios courants.

Certains types d'instances créés sur le Système Nitro incluent des accélérateurs matériels, conçus à la fois par AWS et par des tiers (tels que des processeurs graphiques ou GPU). L'Hyperviseur Nitro est également chargé d'attribuer ces périphériques matériels à la machine virtuelle, de traiter les cas d'erreur matérielle et d'exécuter d'autres fonctions qui ne peuvent pas être exécutées via une interface de gestion « out-of-band ».

L'Hyperviseur Nitro ne comporte, de par sa conception, aucune couche logicielle réseau, aucune implémentation de système de fichiers et aucune prise en charge de pilotes de périphériques. L'Hyperviseur Nitro a été conçu pour inclure uniquement les services et fonctionnalités strictement nécessaires à sa tâche : il ne comprend ni interface de ligne de commande (« shell ») ni aucun mode d'accès interactif. La petite taille et la relative simplicité de l'Hyperviseur Nitro constituent un avantage de sécurité significatif par rapport aux hyperviseurs classiques.

Le code de l'Hyperviseur Nitro est un composant géré et signé de manière cryptographique, semblable à un microprogramme (firmware), stocké sur le stockage local chiffré et rattaché au Contrôleur Nitro. Il est ainsi chaîné à la base de confiance matérielle (root-of-trust) du Contrôleur Nitro. Lorsqu'un système est configuré pour utiliser l'Hyperviseur Nitro, le Contrôleur Nitro charge directement sur la carte mère du système une copie du code de l'hyperviseur, dont l’authenticité a été vérifiée, comme cela se ferait pour un microprogramme.

> **Note**
>
> Le mécanisme « d'injection » de l'hyperviseur utilise un périphérique NVMe en lecture seule, fourni par le Contrôleur Nitro à la carte mère, en tant que lecteur de démarrage du système.

Le transfert du traitement des données et de la virtualisation des E/S à du matériel dédié, ainsi que la réduction des responsabilités confiées à l'hyperviseur exécuté sur le processeur hôte, sont au cœur de la conception du Système Nitro. Cette architecture ne fournit pas seulement des performances améliorées et une sécurité renforcée grâce à l'isolation, elle permet également de proposer des instances de type « bare metal » d'EC2, l'hyperviseur est ainsi un composant qui n'est pas nécessaire pour fournir la virtualisation des E/S, la gestion du système ou la surveillance.

> **Note**
>
> Le Système Nitro garantit des performances comparables à du « bare metal » en exécutant presque toutes les fonctionnalités du système de virtualisation sur les Cartes Nitro plutôt que sur les processeurs de la carte mère du système hôte. Reportez-vous à la page Performances du Bare Metal avec le système AWS Nitro pour plus d'informations.

La suppression des fonctionnalités non essentielles de l'Hyperviseur Nitro élimine des catégories entières de bogues dont les autres hyperviseurs peuvent être affectés, tels que les attaques réseau à distance ou les augmentations de privilèges liées à des pilotes. Même dans le cas peu probable de la présence d'un bogue dans l'Hyperviseur Nitro qui permettrait d'accéder à du code privilégié, il constitue tout de même un environnement inhospitalier pour tout attaquant potentiel en raison de l'absence de fonctionnalités standard du système d'exploitation comme la ligne de commande, les systèmes de fichiers, les utilitaires courants ou l'accès à des ressources susceptibles de faciliter les accès latéraux au sein de l'infrastructure.

Par exemple, comme nous l'avons déjà souligné, l'Hyperviseur Nitro ne possède aucune couche logicielle réseau et aucun accès au réseau EC2. Au lieu de cela, le Contrôleur Nitro et les autres Cartes Nitro assurent tous les accès de quelque nature que ce soit au réseau extérieur, que le processeur du serveur principal exécute l'Hyperviseur Nitro ou qu'il fonctionne en mode « bare metal ». De plus, comme nous le verrons en détail par la suite, la conception passive des communications de Nitro signifie que toute tentative de « communication » par du code exécuté dans le contexte de l'hyperviseur vers les Cartes Nitro sera refusée et fera l'objet d'une alarme.

Processus de mise à jour de l'Hyperviseur Nitro

ALes mises à jour régulières constituent un aspect crucial du maintien de la sécurité du système. L'Hyperviseur Nitro permet une mise à jour complète du système en direct sans interruption. Lorsqu'une nouvelle version de l'Hyperviseur Nitro est disponible, le code complet de l'hyperviseur en cours d'exécution est remplacé sur-le-champ, tout en préservant les instances EC2 des clients, avec un impact quasi imperceptible sur les performances de ces instances. Ces processus de mise à jour sont conçus de telle sorte qu'à aucun moment les protocoles de sécurité ou les défenses du Système Nitro n'ont besoin d'être mis en pause ou assouplis. Cette fonctionnalité de mise à jour en direct est conçue pour ne pas interrompre les instances des clients, tout en garantissant que non seulement les nouvelles fonctionnalités, mais aussi les mises à jour de sécurité peuvent être appliquées régulièrement et rapidement.

Le découplage entre les temps d'arrêt des instances clients et les mises à jour des composants du Système Nitro évite au service EC2 d'avoir de devoir jongler entre l'expérience client et l'impact potentiel sur la sécurité lors des mises à jour du système, améliorant ainsi la sécurité.

Application à un cas concret : le rattachement d'un volume EBS

Pour avoir une meilleure idée du nombre de composants du Système Nitro qui fonctionnent ensemble, examinons ce qu'il se passe lorsqu'un client effectue un appel à l'API EC2 qui modifie l'état de fonctionnement de son instance EC2 sur un Système Nitro. Nous examinerons en particulier le cas où un client attache un volume EBS crypté existant à une instance en cours d'exécution.

Dans un premier temps, le client utilise l'AWS Command Line Interface (AWS CLI), le SDK AWS ou la AWS Management Console pour appeler la commande `AttachVolume`, en ciblant l'instance choisie. Après avoir vérifié que l'identité IAM du client est authentifiée et autorisée à exécuter la commande `AttachVolume`, l'appel d'API est traité par un ensemble de microservices au sein des plans de contrôle EC2 et EBS. Au final, les services du plan de contrôle font appel à un ensemble défini d'API réseau, fournies par le Contrôleur Nitro, chiffrées et authentifiées, avec les informations requises pour allouer les ressources nécessaires au rattachement du volume. Plusieurs services sont impliqués dans cette opération, chaque microservice prenant en charge des tâches distinctes qui limitent l'étendue de l'accès aux API du Contrôleur Nitro.

Le plan de contrôle EC2 alloue les ressources du périphérique PCIe de la carte Nitro pour EBS qui sont nécessaires aux opérations de lecture et d'écriture sur le volume logique EBS (soit une fonction virtuelle NVMe pour une instance virtualisée, soit une fonction physique NVMe pour une instance « bare metal »). Le plan de contrôle EBS fournit les informations nécessaires pour se connecter aux serveurs EBS hébergeant les données chiffrées du volume sur le réseau, ainsi qu'une copie chiffrée de la clé de données du volume qui est stockée sous forme de métadonnées du volume. La clé de données chiffrée est protégée par une clé AWS KMS présente uniquement dans AWS Key Management Service (AWS KMS). Par conséquent, dans le cadre du processus d'attachement du volume, la clé chiffrée doit être envoyée à AWS KMS pour être déchiffrée.

En supposant que l'identité IAM du client à l'origine de la commande `AttachVolume` soit également autorisée à exécuter une commande Decrypt dans AWS KMS pour la clé AWS KMS en question, la clé de données du volume chiffré sera déchiffrée. L'accès du Système Nitro à cette opération est protégé par AWS KMS Grants et par les sessions d'accès IAM Forward. (Reportez-vous à cette explication des IAM Forward Access Sessions dans le contexte d'Elastic Load Balancing, d'AWS Certificate Manager et d', dans une présentation de Colm MacCárthaigh, vice-président et ingénieur émérite chez AWS.)

Ensemble, ces mécanismes garantissent de manière cryptographique que le Système Nitro n’est autorisé à utiliser une clé d’un client gérée par AWS KMS que lorsque le client a récemment autorisé et authentifié cet accès. Le Système Nitro n'est pas autorisé à utiliser les clés gérées par AWS KMS de manière ponctuelle ou en l'absence d'une autorisation récente du client.

Après avoir été déchiffrée dans AWS KMS et avant d'être envoyée au Contrôleur Nitro à l'aide d'une connexion réseau TLS (Transport Layer Security) chiffrée, AWS KMS chiffre à nouveau la clé de données à l'aide d'une clé publique qui sert d'identité numérique cryptographique pour le serveur hôte Nitro de production en question. Cette clé publique a été envoyée avec la clé de données du volume chiffré par le plan de contrôle EBS à AWS KMS. Par conséquent, en plus du chiffrement de l'intégralité du message en transit par TLS, la clé de données est également chiffrée de manière asymétrique dans le message, c'est-à-dire qu’elle est chiffrée deux fois. Seule la Carte Nitro de cet hôte de production spécifique supportant l'environnement informatique de ce client spécifique possède la clé privée nécessaire pour déchiffrer la clé de données chiffrée. Une fois déchiffrée localement, la clé de données en clair est stockée uniquement dans la mémoire volatile de cette Carte Nitro, pour la durée de l’attachement et de l'utilisation du volume.

La Carte Nitro EBS est maintenant prête à présenter le volume EBS à l'instance EC2 via une connexion PCIe à une interface NVMe. Lorsque l'hôte est configuré pour utiliser l'Hyperviseur Nitro, le Contrôleur Nitro envoie un message via l'interface PCIe pour demander à l'Hyperviseur Nitro d'attribuer la fonction virtuelle NVMe pour ce volume EBS à l'instance EC2 appropriée. L'Hyperviseur envoie ensuite un événement de connexion à chaud du matériel virtuel à la machine virtuelle pour avertir le logiciel système du client qu'un nouveau périphérique de stockage en mode bloc NVMe est disponible. Dans le cas d'une instance « bare metal », la Carte Nitro pour EBS signale un événement de connexion à chaud PCIe directement au processeur du serveur, et le logiciel système du client exécuté sur le processeur gère l'événement de connexion à chaud PCIe du périphérique NVMe, comme il le ferait sur n'importe quel autre serveur.

À ce stade, le système d'exploitation de l'instance client s'exécutant en tant qu'invité virtuel ou en tant qu'instance « bare metal » interagit avec un périphérique NVMe présenté par la Carte Nitro pour EBS via l'interface PCIe. Cette interaction se produit soit sous la forme d'une fonction SR-IOV dans le cas d'instances EC2 virtuelles, soit sous la forme d'une fonction physique PCIe dans le cas d'instances EC2 « bare metal ». Les commandes NVMe envoyées via l'interface PCIe sont traitées par le microprogramme exécuté sur la Carte Nitro pour EBS, qui interagit à son tour avec le service EBS via l'interface réseau intégrée du SoC Nitro. Comme indiqué précédemment, la Carte Nitro EBS est également capable de décharger les opérations cryptographiques des volumes EBS chiffrés au format AES-256 XTS, de sorte que chaque bloc de données client soit entièrement chiffré avant de quitter la Carte Nitro, sans impact sur les performances. Un client peut également choisir d'utiliser

un système de fichiers chiffré au niveau du système d'exploitation afin que toutes les données client soient entièrement cryptées avant d'être écrites ou transmises sur la Carte Nitro pour EBS. Cette approche met en place une couche de chiffrement supplémentaire pour les données EBS, à la fois lors du transit et dans le système de stockage EBS.

L’absence d’accès pour les opérateurs AWS

De par sa conception, le Système Nitro n’offre pas d’accès opérateur. Il n'existe aucun mécanisme permettant à un système ou à une personne de se connecter aux hôtes EC2 Nitro, d'accéder à la mémoire des instances EC2 ou d'accéder à des données client stockées sur un stockage d'instance chiffré local ou sur des volumes EBS chiffrés distants. Si un opérateur d’AWS, y compris ceux disposant des privilèges les plus élevés, doit effectuer des travaux de maintenance sur un serveur EC2, il ne peut utiliser qu'un ensemble limité d'API d’administration authentifiées, autorisées, enregistrées et auditées. Aucune de ces API ne permet à un opérateur d'accéder aux données des clients sur le serveur EC2. Comme il s'agit de restrictions techniques conçues, testées et intégrées au Système Nitro lui-même, aucun opérateur AWS ne peut contourner ces contrôles et ces protections.

Comme pour la plupart des décisions d'ingénierie, le choix de concevoir le Système Nitro sans mécanisme d'accès pour les opérateurs nous a obligés à faire des compromis. Dans les rares cas où des problèmes très spécifiques surviennent, l'absence d'accès générique implémenté sur le matériel de production empêche les opérateurs AWS de déboguer sur place. Dans ces rares circonstances, nous devons travailler avec les clients, à leur demande, pour reproduire ces problèmes spécifiques sur du matériel de débogage Nitro, non destiné à la production. Cela peut s'avérer moins pratique que si nos opérateurs pouvaient effectuer le débogage sur place, mais nous sommes convaincus que c'est le meilleur compromis pour nos clients. Cette situation nous oblige également à respecter les normes les plus strictes en matière de qualité et de tests avant la mise en production.

Le principe des communications passives

La conception du système AWS Nitro suit un principe dite de « communication passive ». Cela signifie que, en production, les composants du système n'initient jamais de communication sortante, y compris vers un plan de contrôle, un service de gestion ou un service cloud. Au lieu de cela, un seul service de confiance et spécifiquement durci écoute le réseau, écoute les commandes lancées sur le réseau ou le bus système, agit en fonction de ces commandes, puis renvoie des résultats, le tout via des API bien définies avec un accès restreint. Les deux extrémités de ces voies de communication effectuent également une validation des paramètres afin de garantir que seuls des paramètres valides sont envoyés et reçus.

Ce principe commence par l'Hyperviseur lui-même. Il attend les commandes du Contrôleur Nitro sur un canal privé via PCIe. Il n'initie jamais de communication sortante avec le reste du Système Nitro. Il ne peut pas établir de connexion réseau sortante car, comme indiqué précédemment, il ne possède aucune couche réseau. Si, à un quelconque moment, par le biais d'une série d'actions improbables, l'Hyperviseur Nitro tentait d'établir des communications avec d'autres composants du Système Nitro, cela indiquerait clairement un défaut du microprogramme ou une possible compromission du système, et le service EC2 est conçu pour réagir en conséquence de façon à éviter tout impact et à alerter pour qu’un opérateur intervienne.

> **Note**
>
> En mode « bare metal », aucun hyperviseur ne s'exécute sur le processeur du serveur pour attendre les instructions du Contrôleur Nitro afin de démarrer, arrêter ou réinitialiser le serveur hôte. Dans ce cas, le Contrôleur Nitro contrôle la carte mère via sa connexion BMC privée et la Puce de Sécurité Nitro.

Le modèle de communication passive s’applique également à la couche suivante du Système Nitro. Le Contrôleur Nitro écoute sur un canal réseau sécurisé en attente de commandes authentifiées et autorisées sous la forme d'API spécifiques invoquées par le plan de contrôle EC2. Le Contrôleur Nitro n'initie jamais de communications sortantes sur le réseau du plan de contrôle EC2. Même les fonctionnalités « push » logiques, telles que la publication de métriques CloudWatch pour les instances EC2 exécutées sur l'hôte ou l'envoi des journaux de l'API Nitro au plan de contrôle EC2, sont mises en œuvre sous la forme d'un processus « pull ». Le plan de contrôle interroge régulièrement le Contrôleur Nitro pour récupérer les métriques à l'aide d'API bien définies. Toute tentative de communication sortante depuis le Contrôleur Nitro indiquerait clairement un bogue du

microprogramme ou une possible compromission du système. Le service EC2 est conçu pour réagir en conséquence de façon à éviter tout impact et à alerter l'opérateur pour qu'il intervienne.

Le résultat de ce modèle de communication passive est un degré élevé d'isolation et de sécurité. Comme le fonctionnement normal implique de n'écouter que des messages bien définis, dont les paramètres ont été validés et d'y répondre à l'aide de réponses bien définies et dont les paramètres ont été validés, le système est conçu pour repérer et signaler toute activité qui paraîtrait suspecte. Le système est conçu de telle sorte que, dans le cas peu probable d'un bogue du microprogramme sur la carte mère, il est très probable qu'un adversaire potentiel tentant de s'échapper de la carte mère vers les Cartes Nitro soit détecté, bloqué et signalé. De plus, même dans le cas extrêmement improbable où un adversaire potentiel parviendrait tout de même à s'échapper de la carte mère et à accéder d'une manière ou d'une autre aux Cartes Nitro, la conception du Système Nitro fait à nouveau en sorte que toute tentative de cet adversaire de s'échapper des cartes Nitro serait très probablement détectée, bloquée et signalée, précisément pour les mêmes raisons. Ces multiples couches de défense protègent non seulement le service EC2 lui-même, mais également tous les clients exécutant des environnements et des applications au sein du système EC2.

La gestion des mises à jour du Système Nitro

Le logiciel et le micrologiciel qui sous-tendent le Système Nitro sont développés par des équipes d'ingénieurs réparties dans le monde entier. Toutes les configurations et modifications de code liées au Système Nitro sont soumises à un examen et à une approbation multipartites, ainsi qu'à des déploiements échelonnés dans les environnements de test comme dans les environnements de production. Le développement du logiciel commence par des documents de conception et des révisions, puis passe par plusieurs revues du code. Un examen de sécurité est effectué à la fois par l'équipe de sécurité indépendante d'AWS et par l'équipe d'ingénierie d'Amazon EC2 dans le cas de modifications ou fonctionnalités importantes.

Toutes les modifications sont examinées par au moins un membre supplémentaire de l'équipe d'ingénierie ou une partie prenante, ainsi que par un ingénieur doté d'une longue expérience en matière d'EC2 et membre de notre programme Change Management Bar Raiser. Outre l'examen par un expert, tous les enregistrements de code doivent passer une batterie de contrôles automatisés de qualité et de sécurité qui ne peuvent être contournés et qui s'exécutent automatiquement sous le contrôle d'un service centralisé d'intégration (build) garantissant le respect des meilleures pratiques de déploiement, y compris une surveillance et une restauration appropriées.

Une fois que toutes les vérifications et approbations du code sont terminées et que tous les contrôles automatisés ont été réussis, notre processus de déploiement automatique des packages prend le relais. Dans le cadre de ce pipeline de déploiement automatisé, des binaires sont créés et les équipes exécutent des tests de validation de de la sécurité de bout en bout. Si un quelconque type de validation échoue, le processus de déploiement est interrompu jusqu'à ce que le problème soit résolu.

Les fichiers binaires des logiciels et des microprogrammes sont signés cryptographiquement à l'aide d'une clé privée asymétrique uniquement accessible via le pipeline automatisé, qui enregistre toutes les activités de signature.

Les logiciels et microprogrammes signés sont ensuite déployés dans la flotte d'Amazon EC2 par un système de déploiement dédié, configuré pour suivre une politique et un calendrier de déploiement définis. Les modifications se déploient par vagues dans les zones de disponibilité et les régions. Les déploiements sont surveillés pour s'assurer que seules les versions logicielles qui fonctionnent comme prévu restent déployées et que tout comportement anormal d'une version logicielle déclenche l'annulation automatique de son déploiement.

> **Note**
>
> Consultez la bibliothèque Amazon Builder pour en savoir plus sur la façon dont Amazon crée et exploite des logiciels, en particulier : Automatisation de déploiements sécurisés sans intervention, par Clare Liguori, Sr. Principal Engineer chez AWS, Être plus rapide avec la distribution continue, par Mark Mansour, Senior Manager of Software Development chez AWS, et Exécuter des annulations sûres pendant les déploiements, par Sandeep Pokkunuri, Sr. Principal Engineer chez AWS.

L'approche d’EC2 pour prévenir les attaques par canaux auxiliaires

Depuis sa création, EC2 a toujours adopté une approche conservatrice en matière de conception et d'exploitation de systèmes mutualisés sécurisés pour ses clients. Notre approche de conception privilégie les abstractions simples et fiables, qui fournissent une isolation robuste entre les domaines de sécurité et limitent le partage des ressources système critiques entre les clients. AWS conçoit ses systèmes non seulement pour fournir une défense en profondeur contre les menaces de sécurité connues, mais également pour éviter les problèmes de sécurité potentiels qui ne sont pas associés à des techniques d'exploitation pratiques connues. Outre les mécanismes de sécurité soigneusement testés et bien établis que nous utilisons en production, AWS participe activement à des recherches de pointe sur la sécurité afin de garantir non seulement que nous restons à jour, mais aussi que nous surveillons activement les problèmes de sécurité pour le compte de nos clients.

Les recherches et les divulgations dans le domaine des canaux auxiliaires microarchitecturaux publiées ces dernières années ont attiré l'attention sur cette question. Les canaux auxiliaires sont des mécanismes susceptibles de compromettre les informations secrètes traitées par un système informatique grâce à l'analyse de données indirectes collectées à partir de ce système. Le temps nécessaire à un système pour le traitement d’une donnée d’entrée est un exemple de telles données indirectes. Dans certains cas, bien qu'un système ne révèle jamais directement une donnée secrète, un tiers peut être en mesure de déterminer la valeur de cette donnée grâce à une analyse minutieuse des différences de temps de traitement de données d'entrées soigneusement sélectionnées.

> **Note**
>
> Un exemple simple d'un tel scénario serait un programme qui reçoit un mot de passe sous la forme d'une chaîne de caractères en entrée et vérifie si cette chaîne correspond à une valeur secrète. Ce programme analyse la chaîne de caractères fournie, caractère par caractère, en comparant chaque caractère au caractère correspondant de la valeur secrète et renvoie une erreur dès qu'il rencontre une différence. Bien que le programme ne communique jamais au demandeur la valeur de la chaîne secrète, le programme « divulgue » des informations sur celle-ci sous la forme d'un temps de réponse différent pour une entrée qui commence par un ou plusieurs des mêmes caractères que la chaîne secrète et pour une entrée qui n'en contient pas. Grâce à un processus d'essais et d'erreurs systématiques, un observateur peut

être en mesure de mesurer le temps nécessaire pour répondre à certaines entrées afin de déterminer la valeur de la chaîne secrète, caractère par caractère.

Le déploiement minutieux de contre-mesures telles que celles utilisées par s2n-tls, le protocole de chiffrement SSL/TLS open source d'AWS, est un moyen de se protéger contre ces formes de divulgation de données par des canaux auxiliaires.

> **Note**
>
> s2n-tls intègre des contre-mesures d'équilibrage temporel afin de s'assurer que le timing du processus ne soit influencé que de manière négligeable par les valeurs secrètes, et le démontre par des méthodes formelles. Par conséquent, aucun comportement temporel observable par un attaquant ne dépend de celles-ci. Pour en savoir plus sur ces contre-mesures dans s2n-tls et les méthodes formelles, consultez l'article SideTrail: Verifying Time-Balancing of Cryptosystems.

Les canaux auxiliaires microarchitecturaux impliquent spécifiquement la manipulation du comportement de bas niveau du processeur d'un système dans certaines circonstances, afin de permettre à un processus exécuté sur ce système de déterminer indirectement la valeur de données secrètes via des ressources du système comme les caches, les buffers internes et d'autres sources de données actives auxquelles il n'est pas permis d'accéder directement. Ces canaux auxiliaires sont essentiellement centrés sur le partage de l'accès aux ressources matérielles de bas niveau entre deux systèmes.

AWS adopte une approche conservatrice en matière d'isolation des tenants du service EC2, décrite dans les sections suivantes, conçue de telle sorte que les instances clients ne puissent jamais partager des ressources système telles que le cache L1/L2 ou des threads exécutés sur le même cœur de processeur. Ce choix de conception fondamental exclut la possibilité de fuite de données provenant des instances clients par le biais de canaux auxiliaires microarchitecturaux, qui reposent sur un accès partagé à ces ressources entre les tenants.

Les protections contre les canaux auxiliaires dans le cadre plus large du service EC2

Toutes les instances EC2 incluent des protections robustes contre les canaux auxiliaires. Cela inclut à la fois les instances basées sur le Système Nitro et sur l'Hyperviseur Xen. Bien que cette section traite des protections du Système Nitro, celles-ci sont également présentes dans les instances EC2 basées sur Xen.

Les types d'instances EC2 virtualisées se répartissent en deux catégories :

- Instances à performances fixes instances, dans lesquelles les ressources de processeur et de mémoire sont préallouées et dédiées à une instance virtualisée pendant toute la durée de vie de cette instance sur l'hôte.
- Instances aux performances évolutives, dans lesquelles les ressources du processeur et de la mémoire peuvent être surutilisées afin de prendre en charge un plus grand nombre d'instances virtualisées exécutées sur un serveur et, par conséquent, d'offrir aux clients un coût relatif par instance réduit pour les applications dont l'utilisation du processeur est faible à modérée. Reportez-vous à la section Burstable performance instances de notre documentation.

Dans les deux cas, la conception et la mise en œuvre de l'Hyperviseur Nitro incluent de multiples protections contre les canaux auxiliaires potentiels.

Pour les instances à performances fixes, l'affectation de ressources fournit à la fois une protection naturelle contre les canaux auxiliaires et des performances supérieures par rapport aux autres hyperviseurs. Par exemple, 16 processeurs virtuels (huit cœurs, chaque cœur fournissant deux threads) ainsi que 32 Go de mémoire sont alloués à une instance c5.4xlarge. Lorsqu'une instance est lancée, le plan de contrôle EC2 demande au Contrôleur Nitro d'allouer les ressources de processeur, de mémoire et d'E/S nécessaires pour prendre en charge l'instance.

L'Hyperviseur Nitro reçoit l’ordre du Contrôleur Nitro pour allouer l’ensemble des cœurs physiques et de la mémoire demandés à l'instance. Ces ressources matérielles sont attachées à cette instance particulière. Les cœurs du processeur ne sont pas utilisés pour exécuter les environnements d'autres clients et aucune page mémoire d'instance n'est partagée de quelque manière que ce soit entre les instances, contrairement à de nombreux hyperviseurs qui peuvent consolider des données et/ou des pages d'instructions dupliquées afin de préserver la mémoire physique.

Même sur de très petites instances Nitro EC2, aux ressources limitées, les cœurs de processeur ne sont jamais partagés simultanément entre deux instances clientes via le multithreading simultané

(SMT). Les instances client se voient attribuer des multiples de deux vCPU ou représentant deux threads d'un seul cœur pour les processeurs utilisant le protocole SMT, ou bien un unique vCPU pour les configurations de processeurs associant un seul thread par coeur (comme les processeurs AWS Graviton). L'absence de partage de cœurs signifie qu'aucun cache de niveau 1 ou de niveau 2 ni aucune autre ressource spécifique au cœur, telle que l'exécution spéculative ou l'état d'économie d'énergie, n’est partagée.

Certaines tailles d'instance peuvent partager certaines lignes de dernier niveau de cache de manière non simultanée. Bien qu'il soit possible d'utiliser l'amorçage et le sondage des lignes de dernier niveau de cache comme signal à très faible bande passante entre des processus coopérants, cela ne constitue pas en pratique un canal auxiliaire. En effet, du fait de leur fonction, seules les données rarement consultées sont référencées dans les lignes de cache de dernier niveau. Les canaux auxiliaires nécessitent généralement un nombre d'échantillons très important et statistiquement pertinent pour surmonter le bruit présent dans les systèmes.

Aucune attaque praticable n’est disponible lorsque, comme dans EC2, les pages mémoire ne sont pas partagées entre des serveurs virtuels. À ce jour, toutes les attaques par canal auxiliaire microarchitectural s’appuient soit des cœurs partagés via SMT, soit des caches L1/L2 partagés, soit d'autres attributs de bas niveau tels que des unités à virgule flottante. Les mesures de réduction des risques d’attaque par canal auxiliaire sont très efficaces dans EC2, car ces ressources ne sont jamais partagées dans un environnement EC2 Nitro.

> **Note**
>
> EC2 expose avec précision la topologie sous-jacente du processeur au niveau du matériel, y compris le cache de dernier niveau (généralement L3) et les informations d'accès non uniforme à la mémoire (NUMA), directement via les instances. Il est donc possible pour les clients de déterminer, en inspectant la taille de l'instance, le nombre de cœurs de processeur nécessaires pour « remplir » exactement un ou plusieurs segments du processeur qui partagent un cache L3, et de déterminer ainsi si une instance donnée partage ou non un cache L3 avec une autre instance. Les topologies de partage du cache L3 diffèrent selon la conception du processeur et peuvent être partagées entre un cœur, un ensemble de processeurs ou un ensemble de cœurs en fonction de l'architecture du processeur. Par exemple, dans un système EC2 Intel classique à deux sockets, une taille d'instance égale à la moitié de la taille la plus importante remplira un socket complet et ne partagera pas le cache L3 avec une autre instance.

S’agissant des instances EC2 avec performances évolutives (par exemple, T3, T3a et T4g), elles peuvent utiliser des ressources de processeur et de mémoire dépassant les ressources physiques. Les ressources du processeur nécessaires à l'exécution d'instances aux performances évolutives sont planifiées en fonction d'une allocation basée sur les crédits. Dans cette famille d'instances peu coûteuses mais relativement performantes, même les plus petits types d'instances fournissent toujours aux clients un minimum de deux processeurs virtuels (un cœur, deux threads) sur des processeurs utilisant le protocole SMT.

Il est toutefois possible que deux instances EC2 avec performances évolutives s'exécutent de manière séquentielle (et non simultanément) sur le même cœur. Il est également possible que les pages de mémoire physique soient réutilisées, remappées et échangées en tant que pages de mémoire virtuelle. Cependant, même ces instances ne partagent jamais le même cœur en même temps, et les pages de mémoire virtuelle ne sont jamais partagées entre les instances.

L'Hyperviseur Nitro utilise un certain nombre de stratégies de sécurité à chaque changement de contexte entre les instances afin de garantir que tous les états de l'instance précédente sont supprimés avant d'exécuter une autre instance sur le ou les mêmes cœurs. Cette pratique réduit efficacement les risques d'éventuelles attaques par canal auxiliaire.

Pour les instances EC2 avec performances évolutives, le Système Nitro peut utiliser des techniques de gestion de la mémoire telles que la réutilisation, le remappage ou le remplacement de la mémoire physique sous forme de pages de mémoire virtuelle, mais le système est conçu de telle sorte que les pages de mémoire virtuelle ne soient jamais partagées entre les instances, afin de maintenir une isolation stricte.

Enfin, les instances avec performances évolutives, qu'elles soient prises pour cible ou que l’on cherche à obtenir des données par le biais de techniques de canaux auxiliaires, peuvent être reprogrammées sur des cœurs différents de ceux utilisés précédemment, ce qui limite encore la possibilité de tout type d’attaque temporelle.

Les avantages supplémentaires du Système Nitro en matière de canaux auxiliaires

Outre les protections fournies par EC2 pour Xen et Nitro, la conception du Système Nitro et de l'Hyperviseur Nitro présente des avantages qui ne sont pas évidents de prime abord, mais qui sont très importants pour lutter contre les risques liés aux canaux auxiliaires. Alors que certains hyperviseurs ont dû mettre en œuvre d’importantes modifications pour isoler l'espace d'adressage et réduire les risques d'attaque par canal auxiliaire de type L1 Terminal Fault (par exemple, voir l’article

Hyper-V HyperClear Mitigation for L1 Terminal Fault), le Système Nitro, du fait de l'isolation entre l'espace d'adressage virtuel de l'Hyperviseur Nitro et la mémoire allouée aux instances clients, fournit une immunité naturelle à cette attaque.

Nous avons également mis à profit les enseignements tirés de la conception du Système Nitro pour atténuer les menaces émergentes liées aux attaques par canaux auxiliaires microarchitecturaux dans la version communautaire de l'Hyperviseur Xen. Reportez-vous à la présentation Running Xen Without a Direct Map.

Comme il a été indiqué précédemment, le Système Nitro réduit considérablement la quantité de code du système EC2 exécuté sur le processeur du serveur physique, ce qui diminue énormément la surface d'attaque de l'hyperviseur et isole le traitement des données d'E/S des clients du reste du système. Le code AWS nécessaire pour fournir les fonctionnalités d'E/S définies par logiciel d'EC2 ne s'exécute pas sur les mêmes processeurs que ceux qui exécutent les environnement des clients.

Cette isolation ainsi que l'utilisation de matériel dédié signifient que le traitement des données client dans les sous-systèmes d'E/S est isolé au niveau des fonctions matérielles et ne réside pas dans la mémoire hôte, les caches des processeurs ou d'autres mémoires tampons internes, contrairement aux logiciels de virtualisation traditionnels qui mélangent ces données car ils s'exécutent sur les processeurs partagés sur l'hôte.

Toutes ces protections sont rendues possibles par l'implication d'AWS dans la recherche en sécurité, AWS menant souvent la recherche et la découverte de problèmes et coordonnant les mesures visant leur atténuation.

Les Enclaves Nitro

Les Enclaves Nitro sont une fonctionnalité du Système Nitro qui permet aux clients de diviser leurs environnements en composants distincts qui n'ont pas nécessairement totalement confiance entre eux. Il s'agit également d'un moyen d'exécuter du code de confiance et de traiter des données auxquelles les administrateurs d'instances EC2 du client n'ont pas accès. Nous n'abordons pas les caractéristiques et les avantages des Enclaves Nitro dans le contexte du présent article, mais les points suivants méritent toutefois d'être soulignés.

Une Enclave Nitro hérite de la même isolation et des mêmes mesures de mitigation des risques liés aux canaux auxiliaires que toutes les autres instances EC2 exécutées sur le même processeur de serveur. L'instance parent doit allouer un nombre fixe de processeurs virtuels (la quantité minimale étant égale à un cœur complet) ainsi qu'un nombre fixe de pages mémoire. Cet ensemble fixe de

ressources de processeur et de mémoire est soustrait à l'instance parent (à l'aide de la fonctionnalité de « déconnexion à chaud des ressources matérielles » prise en charge dans les noyaux Linux et Windows) puis utilisé par l'Hyperviseur Nitro pour créer un autre environnement de machine virtuelle indépendant, entièrement protégé, dans lequel exécuter l'image Nitro Enclave.

Toutes les protections décrites ci-dessus sont automatiquement mises en place lors de l'utilisation de Nitro Enclaves, car il n'y a aucun partage de cœur ou de mémoire avec l'instance parent.

Réflexions finales sur les canaux auxiliaires

En résumé, la conception de Nitro et de la plate-forme EC2 offrent des mesures très efficaces pour prévenir la possibilité d'attaques par canal auxiliaire, notamment la suppression de l'accès partagé au processeur et aux ressources mémoire entre les instances. En outre, les clients ont toujours la possibilité de choisir de ne pas provisionner leurs instances sur les mêmes hôtes que les instances appartenant à d'autres clients. Enfin, la participation d'AWS aux groupes de travail du secteur sur les vulnérabilités pour Linux, KVM, Xen et d'autres technologies clés, ainsi que les capacités de mise à jour en direct du Système Nitro, permettront à AWS de réagir rapidement si de nouveaux défis sont identifiés et de protéger les clients contre les nouvelles menaces qui apparaissent, le tout sans perturber leurs applications. Par exemple, AWS a fait partie du petit groupe de sociétés qui ont travaillé sur les menaces Spectre et Meltdown avant leur divulgation publique, et a ainsi pu réduire tous les risques liés à son infrastructure avant publication.

> **Note**
>
> Les clients peuvent choisir de ne pas partager les serveurs physiques avec d'autres clients en utilisant les fonctionnalités « Instances dédiées » ou « Hôtes dédiés » d'EC2. Il s’agit de stratégies de placement d'instances qui permettent à un client d'être le seul à un moment donné à disposer d'instances sur un hôte physique EC2 donné. Reportez-vous à la section Hôtes dédiés d'Amazon EC2.

Nous continuons à travailler avec des partenaires clés comme Intel, AMD et ARM odans le domaine de la recherche en matière de sécurité matérielle et de réponse coordonnée aux vulnérabilités, et nous continuons à innover en matière d'isolation informatique. La solution open source Firecracker VMM en est un exemple. Elle permet à des conteneurs sans serveur et à des services tels que AWS Fargate et AWS Lambda de bénéficier de la sécurité, de l'isolation et de la cohérence de la virtualisation sans sacrifier la vitesse, la flexibilité et les performances dont les clients ont besoin.

> **Note**
>
> Firecracker est une technologie de virtualisation spécialement conçue pour créer et gérer de façon sécurisée des services multi-tenant d'exécution de conteneurs et de fonctions. Firecracker est un moniteur de machine virtuelle qui gère les environnements informatiques dans des micromachines virtuelles légères. Il met en œuvre un modèle de périphérique minimal qui exclut toutes les fonctionnalités non essentielles et réduit la surface d'attaque de la microVM. Outre les optimisations de sécurité et d'isolation qu'il met en œuvre, Firecracker permet également de réduire le temps de démarrage (création de l'espace utilisateur ou du code d'application en 125 ms seulement) et la charge mémoire (5 Mo par micromachine virtuelle).

Les problèmes liés aux canaux auxiliaires constituent un domaine de recherche en constante évolution, ce qui entraîne des innovations et le développement de nouvelles mesures de protection. Nous sommes convaincus que l'expertise approfondie d'AWS et son intérêt continu pour ce domaine sont de nature à protéger efficacement les clients contre ces menaces présentes et futures.

> **Note**
>
> Reportez-vous à cette présentation sur les problèmes liés aux canaux auxiliaires par Eric Brandwine, vice-président et ingénieur émérite chez AWS. Il y parle de la transition de Xen vers Nitro (à 42'40 dans la vidéo) et des avantages qui en découlent, et souligne que ce sujet est devenu semblable à celui de la cryptographie, où l'approche la plus raisonnable consiste à s'appuyer sur des experts chevronnés et à réutiliser leurs travaux (à 49'29).

La sécurité du Système Nitro dans son environnement

Les fonctionnalités du Système Nitro décrites dans ce document s'inscrivent dans le contexte de l'ensemble des contrôles robustes mis en place par AWS pour maintenir la sécurité et la protection des données dans le Cloud AWS. Dans cette section, nous présentons un aperçu de haut niveau des pratiques d'AWS en matière de sécurité et de conformité d'AWS. En tant que client d'AWS, vous bénéficiez de l'ensemble des meilleures pratiques issues des politiques de sécurité d'AWS, de l'architecture et des processus opérationnels conçus pour répondre aux besoins de nos clients les plus exigeants en matière de sécurité.

Les environnements d'AWS sont audités en permanence, avec des certifications délivrées par des organismes d'accréditation de toutes les régions du monde et de tous les secteurs d'activité. AWS Outposts offre également la possibilité aux clients d'exécuter dans leurs propres installations, s'ils le souhaitent, les services AWS de calcul, de stockage, de base de données et d'autres services localement, sur du matériel intégrant le Système Nitro, dans leurs propres installations.

La sécurité des infrastructures

La sécurité chez AWS commence par notre infrastructure de base : le matériel, les logiciels, le réseau et les installations qui exécutent les services du Cloud AWS. Conçue sur mesure pour le cloud et conçue pour répondre aux exigences de sécurité les plus strictes au monde, notre infrastructure est surveillée 24 heures sur 24, 7 jours sur 7 pour garantir la confidentialité, l'intégrité et la disponibilité des données de nos clients. Avec AWS, vous pouvez vous appuyer sur l'infrastructure mondiale la plus sécurisée, en sachant que vous êtes toujours propriétaire de vos données clients, ce qui inclut la possibilité de les chiffrer, de les déplacer et de gérer leur conservation.

L'accès physique

L'accès physique aux centres de données AWS est strictement contrôlé, à la fois sur tout le périmètre du site et aux points d'entrée des bâtiments, par un personnel de sécurité professionnel utilisant la vidéosurveillance, l'authentification biométrique et à deux facteurs, des systèmes de détection des intrusions et d'autres moyens électroniques. Le personnel habilité doit passer avec succès au moins à deux reprises l'authentification multifactorielle pour pouvoir accéder aux étages informatiques. Tous les visiteurs sont tenus de présenter une pièce d'identité et sont introduits et escortés en permanence par du personnel autorisé. AWS n'autorise l'accès aux centres de données et la diffusion d'informations qu'au personnel et aux sous-traitants en ayant légitimement besoin dans le cadre de leurs activités professionnelles.

Lorsqu'un employé n'a plus besoin de tels privilèges pour remplir ses fonctions, son accès est immédiatement révoqué, même s'il fait toujours partie d'Amazon ou d'Amazon Web Services. Tous les accès physiques aux centres de données par le personnel d'AWS sont systématiquement consignés et audités.

La destruction des médias

Les périphériques de stockage multimédia utilisés pour stocker les données des clients sont classés par AWS comme critiques. AWS respecte des normes rigoureuses concernant l'installation, l'utilisation et enfin la destruction des périphériques lorsqu'ils ne sont plus utilisés. Lorsqu'un périphérique de stockage a atteint la fin de sa durée de vie, AWS met le support hors service à l'aide des techniques détaillées dans la norme NIST 800-88. Les supports qui stockaient des données client restent sous le contrôle d'AWS tant qu'ils n'ont pas été mis hors service en toute sécurité.

La protection des données

Toutes les données transitant sur le réseau mondial AWS qui interconnecte nos centres de données et nos régions sont automatiquement chiffrées au niveau de la couche physique avant d'être transmises entre nos installations sécurisées. Des couches de chiffrement supplémentaires existent également, par exemple le trafic d'appairage VPC interrégional et les connexions TLS client ou service-à-service. Nous fournissons des outils qui vous permettent de chiffrer facilement vos données client en transit comme au repos, pour garantir que seuls les utilisateurs autorisés peuvent y accéder, en utilisant des clés que vous contrôlez et gérées dans AWS KMS, ou en gérant vos propres clés de chiffrement avec AWS CloudHSM à l'aide de HSM certifiés FIPS 140-2 de niveau 3.

Nous vous offrons également le contrôle et la visibilité dont vous avez besoin pour vous conformer aux lois et réglementations régionales et locales en matière de confidentialité des données. La conception de notre infrastructure mondiale vous permet de choisir les régions dans lesquelles se trouvent physiquement les vos données client, ce qui vous permet de respecter vos exigences en matière de localisation des données.

Conclusion

Le Système AWS Nitro offre un ensemble unique de fonctionnalités qui lui permettent d’accueillir les applications les plus sensibles dans un environnement cloud multilocataire à grande échelle. Ces fonctionnalités sont basées sur l'investissement d'AWS dans les puces et les microprogrammes associés afin de créer une pile de virtualisation spécifiquement adaptée au besoins du cloud. Depuis le début de l’année 2018, tous les nouveaux types d'instances Amazon EC2 sont basés sur le système AWS Nitro, offrant aux clients tous les avantages de sécurité et autres décrits dans ce document. À la lumière de ces investissements technologiques importants et de l'excellent bilan d'AWS en matière d'isolation des environnements, les clients peuvent utiliser en toute confiance les services cloud d’AWS pour exécuter leurs applications les plus sensibles.

Contributeurs

Ont contribué à ce document :

- J.D. Bean, Principal Security Architect, Amazon EC2
- Mark Ryland, Director, AWS Office of the CISO
- Matthew S. Wilson, Vice President / Distinguished Engineer, Amazon Web Services
- Colm MacCárthaigh, Vice President / Distinguished Engineer, Amazon Web Services
- Benjamin Serebrin, Principal Software Engineer, Amazon EC2

Révisions du document

Pour être informé des modifications apportées à ce livre blanc, inscrivez vous sur ce flux RSS.

Modification	Description	Date
Traduction	Traduction en Français	March 31, 2023
Publication initiale	Publication du Livre Blanc.	November 18, 2022

Avertissement

Il incombe aux clients de procéder à leur propre évaluation indépendante des informations contenues dans ce document. Ce document : (a) est fourni à titre informatif uniquement, (b) couvre les offres et pratiques actuelles des produits AWS qui peuvent être modifiées sans préavis, et (c) ne crée aucun engagement ou aucune garantie de la part d'AWS et de ses sociétés apparentées, fournisseurs ou concédants de licence. Les produits ou services AWS sont fournis « tels quels », sans garantie, engagement ou condition d'aucune sorte, qu'elle soit expresse ou implicite. Les responsabilités et obligations d'AWS vis-à-vis de ses clients sont régies par les contrats AWS. Le présent document ne modifie ni ne fait partie d'aucun contrat entre AWS et ses clients.