

Pilar Segurança



Pilar Segurança: AWS Well-Architected Framework

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Resumo e introdução	1
Introdução	1
Fundamentos de segurança	3
Princípios de design	3
Definição	4
Responsabilidade compartilhada	4
Governança	6
Gerenciamento e separação de contas da AWS	8
SEC01-BP01 Separar as workloads usando contas	9
SEC01-BP02 Proteger as propriedades e o usuário raiz das contas	13
Operar workloads com segurança	18
SEC01-BP03 Identificar e validar objetivos de controle	20
SEC01-BP04 Manter-se a par das ameaças e recomendações de segurança	22
SEC01-BP05 Reduzir o escopo do gerenciamento de segurança	24
SEC01-BP06 Automatizar a implantação de controles de segurança padrão	27
SEC01-BP07 Identificar ameaças e priorizar mitigações com o uso de um modelo de ameaça	30
SEC01-BP08 Avaliar e implementar regularmente novos serviços e recursos de segurança	35
Gerenciamento de identidade e acesso	37
Gerenciamento de identidades	37
SEC02-BP01 Usar mecanismos de login fortes	38
SEC02-BP02 Usar credenciais temporárias	41
SEC02-BP03 Armazenar e usar segredos com segurança	44
SEC02-BP04 Contar com um provedor de identidades centralizado	50
SEC02-BP05 Fazer a auditoria e a alternância periódica das credenciais	54
SEC02-BP06 Utilizar grupos de usuários e atributos	57
Gerenciamento de permissões	60
SEC03-BP01 Definir requisitos de acesso	62
SEC03-BP02 Conceder acesso com privilégio mínimo	64
SEC03-BP03 Estabelecer processo de acesso de emergência	68
SEC03-BP04 Reduzir as permissões continuamente	76
SEC03-BP05 Definir barreiras de proteção de permissões para sua organização	78
SEC03-BP06 Gerenciar o acesso com base no ciclo de vida	82

SEC03-BP07 Analisar o acesso público e entre contas	84
SEC03-BP08 Compartilhar recursos com segurança em sua organização	87
SEC03-BP09 Compartilhar recursos com segurança com terceiros	92
Detecção	97
SEC04-BP01 Configurar registro em log de serviço e aplicação	98
Orientação de implementação	10
Recursos	12
SEC04-BP02 Capturar logs, descobertas e métricas em locais padronizados	103
Orientações para a implementação	10
Etapas da implementação	11
Recursos	12
SEC04-BP03 Correlacionar e enriquecer os alertas de segurança	107
Orientações para a implementação	10
Recursos	12
SEC04-BP04 Iniciar a correção de recursos irregulares	110
Orientações para a implementação	10
Recursos	12
Proteção de infraestrutura	114
Proteção de redes	115
SEC05-BP01 Criar camadas de rede	116
SEC05-BP02 Controlar o fluxo de tráfego dentro das camadas de rede	119
SEC05-BP03 Implementar proteção baseada em inspeção	122
SEC05-BP04 Automatizar a proteção da rede	125
Proteção da computação	128
SEC06-BP01 Fazer o gerenciamento de vulnerabilidades	129
SEC06-BP02 Provisionar computação por meio de imagens reforçadas	132
SEC06-BP03 Reduzir o gerenciamento manual e o acesso interativo	135
SEC06-BP04 Validar a integridade do software	138
SEC06-BP05 Automatizar a proteção da computação	140
Proteção de dados	144
Classificação de dados	144
SEC07-BP01 Compreender seu esquema de classificação de dados	144
SEC07-BP02 Aplicar controles de proteção de dados com base na confidencialidade dos dados	147
SEC07-BP03 Automatizar a identificação e a classificação	150
SEC07-BP04 Definir o gerenciamento escalável do ciclo de vida dos dados	153

Proteção de dados em repouso	155
SEC08-BP01 Implementar gerenciamento de chaves seguro	157
SEC08-BP02 Aplicar criptografia em repouso	160
SEC08-BP03 Automatizar a proteção de dados em repouso	163
SEC08-BP04 Impor o controle de acesso	167
Proteção de dados em trânsito	169
SEC09-BP01 Implementar o gerenciamento seguro de chaves e certificados	170
SEC09-BP02 Aplicar a criptografia em trânsito	173
SEC09-BP03 Autenticar as comunicações de rede	176
Resposta a incidentes	181
Resposta a incidentes da AWS	181
Elaborar objetivos da resposta da nuvem	182
Preparação	183
SEC10-BP01 Identificar equipes e recursos externos fundamentais	184
SEC10-BP02 Desenvolver planos de gerenciamento de incidentes	187
SEC10-BP03 Prepare recursos forenses	191
SEC10-BP04 Desenvolva e teste manuais de resposta a incidentes de segurança	195
SEC10-BP05 Acesso pré-provisionado	196
SEC10-BP06 Pré-implantação de ferramentas	201
SEC10-BP07 Execute simulações	203
Operações	206
Atividade pós-incidente	207
SEC10-BP08 Estabelecer uma estrutura para aprender com os incidentes	207
Segurança de aplicações	211
SEC11-BP01 Treinar para segurança de aplicações	212
Orientação de implementação	10
Recursos	12
SEC11-BP02 Automatizar o teste durante o ciclo de vida de desenvolvimento e lançamento ...	215
.....	215
.....	215
Orientação de implementação	10
Recursos	12
SEC11-BP03 Realizar teste de penetração regular	218
Orientação de implementação	10
Recursos	12
SEC11-BP04 Análises manuais de código	221

Orientação de implementação	10
Recursos	222
SEC11-BP05 Centralizar serviços para pacotes e dependências	223
Orientação de implementação	10
Recursos	12
SEC11-BP06 Implantar software programaticamente	225
Orientação de implementação	10
Recursos	12
SEC11-BP07 Avaliar regularmente as propriedades de segurança dos pipelines	227
Orientação de implementação	10
Recursos	12
SEC11-BP08 Criar um programa que incorpore a propriedade de segurança nas equipes de workload	229
Orientação de implementação	10
Recursos	12
Conclusão	232
Colaboradores	233
Leitura adicional	234
Revisões do documento	235
Avisos	239

Pilar Segurança: AWS Well-Architected Framework

Data de publicação: 27 de junho de 2024 ([Revisões do documento](#))

O foco deste whitepaper é o pilar Segurança do [AWS Well-Architected Framework](#). Ele contém orientações para ajudá-lo a aplicar melhores práticas e recomendações atuais ao projeto, entrega e manutenção de cargas de trabalho seguras da AWS.

Introdução

O [AWS Well-Architected Framework](#) ajuda você a entender as vantagens e desvantagens das decisões que você toma ao criar cargas de trabalho na AWS. Ao usar o Framework, você aprenderá as práticas recomendadas atualmente de arquitetura para projetar e operar workloads confiáveis, seguras, eficientes, econômicas e sustentáveis na nuvem. Com ele, você consegue avaliar consistentemente sua workload em relação às práticas recomendadas e identificar áreas de melhoria. Acreditamos que ter as workloads bem arquitetadas aumenta muito a probabilidade de sucesso nos negócios.

A estrutura é baseada em seis pilares:

- Excelência Operacional
- Segurança
- Confiabilidade
- Eficiência de performance
- Otimização de custos
- Sustentabilidade

Este documento aborda o pilar de segurança. Você encontrará ajuda para cumprir requisitos empresariais e normativos seguindo as recomendações atuais da AWS. Ele destina-se às pessoas com funções de tecnologia, como diretores de tecnologia (CTOs), diretores de segurança da informação (CSOs/CISOs), arquitetos, desenvolvedores e membros da equipe de operações.

Depois de ler este documento, você compreenderá as recomendações e as estratégias atuais da AWS para projetar arquiteturas de nuvem com foco em segurança. Este documento não fornece detalhes de implementação ou padrões de arquitetura; no entanto, inclui referências a recursos relevantes para essas informações. Ao adotar as práticas deste documento, você poderá criar

arquiteturas que protegem dados e sistemas, controlam o acesso e respondem automaticamente a eventos de segurança.

Fundamentos de segurança

O pilar de segurança descreve como aproveitar as tecnologias de nuvem para proteger dados, sistemas e ativos de uma maneira que possa melhorar sua postura de segurança. Este documento fornece orientações detalhadas sobre as melhores práticas para a arquitetura de sistemas confiáveis na AWS.

Princípios de design

Na nuvem, existem vários princípios que podem ajudá-lo a fortalecer a segurança da workload:

- Implementar uma forte base de identidade: implemente o princípio do privilégio mínimo e separe as tarefas com a autorização apropriada para cada interação com os recursos da AWS. Centralize o gerenciamento de identidades e procure eliminar a dependência de credenciais estáticas de longo prazo.
- Mantenha a rastreabilidade: monitore, alerte e audite ações e alterações em seu ambiente em tempo real. Integre a coleta de logs e métricas aos sistemas para investigar e executar ações automaticamente.
- Aplicar segurança em todas as camadas: Aplique uma abordagem de defesa detalhada com vários controles de segurança. Aplique a todas as camadas (por exemplo, borda da rede, VPC, balanceamento de carga, cada instância e serviço de computação, sistema operacional, aplicação e código).
- Automatizar práticas recomendadas de segurança: Mecanismos de segurança baseados em software automatizados melhoram sua capacidade de ajustar a escala de forma segura, mais rápida e com custos reduzidos. Crie arquiteturas seguras, incluindo a implementação de controles definidos e gerenciados como código em modelos controlados por versão.
- Proteger dados em trânsito e em repouso: classifique seus dados em níveis de sensibilidade e use mecanismos, como criptografia, tokenização e controle de acesso, quando apropriado.
- Manter as pessoas afastadas dos dados: Use mecanismos e ferramentas para reduzir ou eliminar a necessidade de acesso direto ou processamento manual de dados. Isso reduz o risco de erros de processamento ou modificação e erro humano ao manipular dados confidenciais.
- Preparar-se para eventos de segurança: Prepare-se para um incidente tendo políticas e processos de gerenciamento e investigação de incidentes alinhados aos requisitos organizacionais. Execute simulações de resposta a incidentes e use ferramentas com automação para aumentar sua velocidade de identificação, investigação e recuperação.

Definição

A segurança na nuvem é composta por sete áreas:

- [Fundamentos de segurança](#)
- [Gerenciamento de identidade e acesso](#)
- [Detecção](#)
- [Proteção de infraestrutura](#)
- [Proteção de dados](#)
- [Resposta a incidentes](#)
- [Segurança de aplicações](#)

Responsabilidade compartilhada

Segurança e conformidade são responsabilidades compartilhadas entre a AWS e o cliente. Esse modelo compartilhado pode ajudar a reduzir os encargos operacionais do cliente porque a AWS opera, gerencia e controla os componentes do sistema operacional do host e a camada de virtualização, incluindo a segurança física das instalações em que o serviço opera. O cliente assume o gerenciamento e a responsabilidade pelo sistema operacional convidado (inclusive por atualizações e correções de segurança) e por outro software de aplicação associado, bem como pela configuração do firewall dos grupos de segurança fornecido pela AWS. Os clientes devem examinar cuidadosamente os serviços que escolherem, pois suas respectivas responsabilidades variam de acordo com os serviços utilizados, a integração desses serviços ao seu ambiente de TI e as leis e regulamentos aplicáveis. A natureza dessa responsabilidade compartilhada também oferece a flexibilidade e o controle do cliente que permitem a implantação. Conforme mostrado no gráfico a seguir, normalmente essa diferenciação da responsabilidade é mencionada como segurança “da” nuvem versus segurança “na” nuvem.

Responsabilidade da AWS com a “segurança da nuvem” : a AWS é responsável pela proteção da infraestrutura que executa todos os serviços oferecidos na Nuvem AWS. Essa infraestrutura abrange o hardware, o software, as redes e as instalações que executam os serviços da Nuvem AWS.

Responsabilidade do cliente com a “segurança na nuvem” : a responsabilidade do cliente será determinada pelos serviços da Nuvem AWS que ele selecionar. Isso define o volume do trabalho de configuração que o cliente deve executar como parte de suas responsabilidades com a segurança.

Por exemplo, um serviço como o Amazon Elastic Compute Cloud (Amazon EC2) é categorizado como infraestrutura como serviço (IaaS) e, como tal, exige que o cliente execute todas as tarefas necessárias de gerenciamento e configuração de segurança. Os clientes que implantam a instância do Amazon EC2 são responsáveis pelo gerenciamento do sistema operacional convidado (incluindo atualizações e patches de segurança), por todos os utilitários ou software de aplicação que instalem nas instâncias e pela configuração do firewall fornecido pela AWS (chamado de grupo de segurança) em cada instância. Para serviços abstraídos, como o Amazon S3 e o Amazon DynamoDB, a AWS opera a camada de infraestrutura, o sistema operacional e as plataformas, e os clientes acessam os endpoints para armazenar e recuperar dados. Os clientes são responsáveis por gerenciar seus dados (incluindo opções de criptografia), classificar seus ativos e usar as ferramentas do IAM para aplicar as permissões apropriadas.

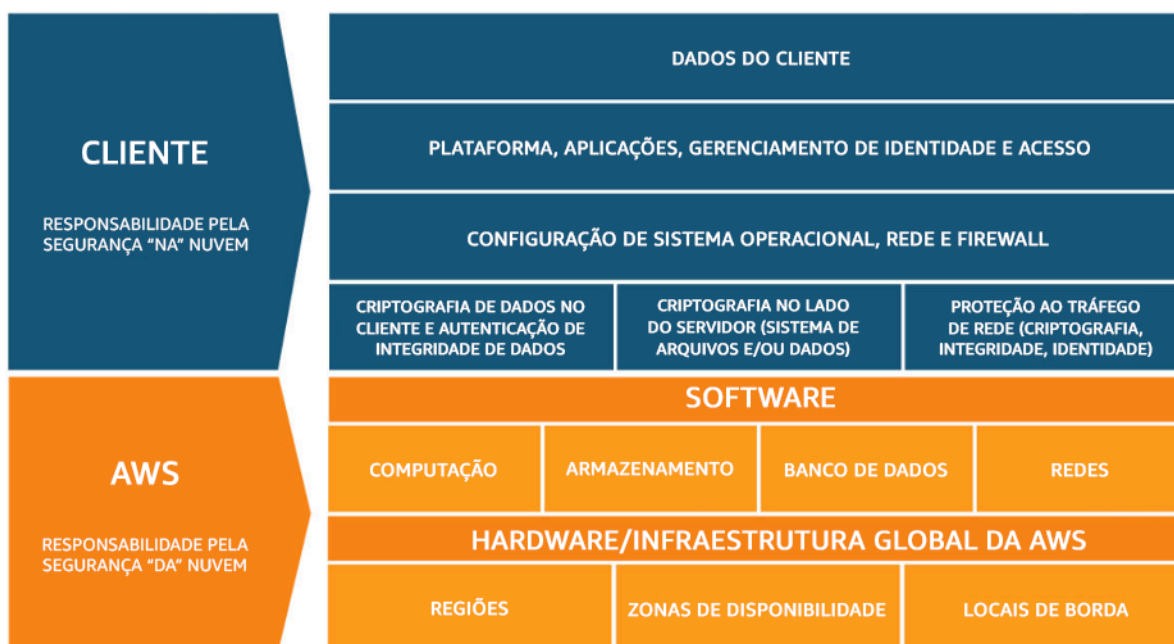


Figura 1: modelo de responsabilidade compartilhada da AWS.

Esse modelo de responsabilidade compartilhada entre o cliente e a AWS também se estende aos controles de TI. Assim como a responsabilidade de operar o ambiente de TI é compartilhada entre a AWS e seus clientes, o gerenciamento, a operação e a verificação dos controles de TI também são compartilhados. A AWS pode ajudar a aliviar a carga de controles operacionais do cliente gerenciando os controles associados à infraestrutura física implantada no ambiente da AWS que pode ter sido gerenciada anteriormente pelo cliente. Já que cada cliente é implantado de forma diferente na AWS, os clientes podem aproveitar a mudança do gerenciamento de determinados controles de TI para a AWS, o que resulta em um (novo) ambiente de controle distribuído. Os clientes podem utilizar a documentação de conformidade e controle da AWS disponível para realizar

procedimentos de avaliação e verificação de controle, conforme necessário. Veja a seguir exemplos de controles gerenciados pela AWS, por clientes da AWS ou por ambos.

Controles herdados : controles que um cliente herda totalmente da AWS.

- Controles físicos e ambientais

Controles compartilhados : controles que se aplicam tanto à camada de infraestrutura quanto às camadas do cliente, mas em contextos ou perspectivas separados. Em um controle compartilhado, a AWS fornece os requisitos para a infraestrutura e o cliente deve fornecer a própria implementação de controle dentro do uso de serviços da AWS. Por exemplo:

- Gerenciamento de aplicação de patches: a AWS é responsável por aplicar patches e corrigir falhas na infraestrutura, mas os clientes são responsáveis por corrigir suas aplicações e sistemas operacionais convidados.
- Gerenciamento de configuração: a AWS mantém a configuração de seus dispositivos de infraestrutura, mas os clientes são responsáveis por configurar os próprios bancos de dados, aplicações e sistemas operacionais convidados.
- Conscientização e capacitação: a AWS capacita os funcionários da AWS, mas os clientes devem capacitar seus próprios funcionários.

Específico para o cliente : controles que são de responsabilidade exclusiva do cliente com base na aplicação que está sendo implantada nos serviços da AWS. Por exemplo:

- Proteção de serviços e comunicações ou segurança de zona, que pode exigir que um cliente roteie ou localize dados em ambientes de segurança específicos.

Governança

A governança de segurança, como um subconjunto da abordagem geral, visa apoiar objetivos de negócios, definindo políticas e objetivos de controle para ajudar a gerenciar riscos. Gerencie os riscos seguindo uma abordagem em camadas para os objetivos de controle de segurança em que cada camada baseia-se na anterior. Compreender o Modelo de Responsabilidade Compartilhada da AWS é a camada mais importante. Esse conhecimento oferece clareza sobre a sua responsabilidade como cliente e o que é herdado da AWS. Um recurso benéfico é o [AWS Artifact](#), que fornece acesso sob demanda aos relatórios de segurança e conformidade da AWS e contratos online selecionados.

Atenda à maioria dos objetivos de controle na próxima camada. É nela que reside a capacidade de toda a plataforma. Por exemplo, essa camada inclui o processo de provisionamento automático de contas da AWS, a integração com um provedor de identidades, como o AWS IAM Identity Center, e os controles de detecção comuns. Alguns dos resultados do processo de governança da plataforma também estão aqui. Quando você quiser começar a usar um novo serviço da AWS, atualize as políticas de controle de serviços (SCPs) no serviço AWS Organizations para fornecer as barreiras de proteção para o uso inicial do serviço. Você pode usar outros SCPs para implementar objetivos comuns de controle de segurança, geralmente chamados de invariantes de segurança. Esses são objetivos de controle ou configuração que você aplica a várias contas, unidades organizacionais ou a toda a organização da AWS. Os exemplos comuns são a limitação das Regiões em que a infraestrutura é executada ou o impedimento da desativação de controles de detecção. Essa camada intermediária também contém políticas codificadas, como regras de configuração ou verificações em pipelines.

A camada superior é onde as equipes de produto atendem aos objetivos de controle. Isso ocorre porque a implementação é feita nas aplicações controladas pelas equipes de produto. Isso pode ser implementar a validação de entrada em uma aplicação ou garantir que a identidade passe entre os microsserviços corretamente. Mesmo que a equipe de produto seja proprietária da configuração, ela ainda pode herdar alguns recursos da camada intermediária.

Independentemente de onde o controle seja implementado, o objetivo continua sendo o gerenciamento de riscos. Uma variedade de estruturas de gerenciamento de riscos se aplica a setores, regiões ou tecnologias específicos. O objetivo principal é destacar o risco com base na probabilidade e na consequência. Esse é o risco inerente. Depois, é possível definir um objetivo de controle que reduza a probabilidade, a consequência ou ambos. Dessa forma, com um controle implementado, você pode ver qual será o risco resultante. Esse é o risco residual. Os objetivos de controle podem ser aplicados a uma ou várias workloads. O diagrama a seguir mostra uma matriz de risco típica. A probabilidade é baseada na frequência de ocorrências anteriores e a consequência é baseada no custo financeiro, de reputação e de tempo do evento.

Probabilidade	Nível de risco				
Muito provável	Baixo	Médio	Alto	Crítica	Crítica
Provável	Baixo	Médio	Médio	Alto	Crítica
Possível	Baixo	Baixo	Médio	Médio	Alto
Improvável	Baixo	Baixo	Médio	Médio	Alto
Muito improvável	Baixo	Baixo	Médio	Médio	Alto
Consequência	Mínima	Baixo	Médio	Alto	Grave

Figura 2: Matriz de probabilidade de nível de risco

Gerenciamento e separação de contas da AWS

Recomendamos que você organize cargas de trabalho em contas separadas e contas de grupo com base na função, nos requisitos de conformidade ou em um conjunto comum de controles, em vez de espelhar a estrutura hierárquica da sua organização. Na AWS, as contas são um limite rígido. Por exemplo, a separação no nível da conta é altamente recomendada para isolar as workloads de produção das de desenvolvimento e teste.

Gerenciar contas de maneira centralizada: o AWS Organizations [automatiza a criação e o gerenciamento de contas da AWS](#), bem como o controle dessas contas depois que elas são criadas. Quando você cria uma conta por meio do AWS Organizations, é importante considerar o endereço de e-mail usado, pois esse será o usuário root que permite que a senha seja redefinida. O Organizations permite agrupar contas em [unidades organizacionais \(UOs\)](#), que podem representar ambientes diferentes com base nos requisitos e na finalidade da carga de trabalho.

Definir controles centralmente: controle o que suas contas da AWS podem fazer, permitindo apenas serviços, Regiões e ações de serviço específicos no nível apropriado. O AWS Organizations permite usar políticas de controle de serviços (SCPs) para aplicar barreiras de proteção de permissão

em nível de organização, unidade organizacional ou conta, que se aplicam a todos os usuários e funções do [AWS Identity and Access Management](#) (IAM). Por exemplo, você pode aplicar uma SCP que restrinja os usuários de iniciar recursos em regiões que você não tenha permitido explicitamente. O AWS Control Tower oferece uma maneira simplificada de configurar e controlar várias contas. Ele automatiza a configuração de contas na organização da AWS, automatiza o provisionamento e aplica [barreiras de proteção](#) (que incluem prevenção e detecção) e fornece um painel para visibilidade.

Configurar serviços e recursos de forma centralizada: o AWS Organizations ajuda você a configurar os [serviços da AWS](#) que se aplicam a todas as suas contas. Por exemplo, você pode configurar o registro em log centralizado de todas as ações executadas em toda a organização usando o [AWS CloudTrail](#) evite que as contas dos membros desativem o registro em log. Você também pode agregar dados centralmente para regras definidas usando o [AWS Config](#), permitindo auditar workloads em busca de conformidade e reagir rapidamente a alterações. O AWS CloudFormation [StackSets](#) permitem que você gerencie centralmente pilhas do AWS CloudFormation entre contas e UOs na sua organização. Isso permite provisionar automaticamente uma nova conta para atender aos seus requisitos de segurança.

Use o recurso de administração delegada dos serviços de segurança para separar as contas usadas para gerenciamento da conta de cobrança organizacional (gerenciamento). Vários serviços da AWS, como GuardDuty, Security Hub e AWS Config, oferecem compatibilidade com integrações com o AWS Organizations, incluindo a designação de uma conta específica para funções administrativas.

Práticas recomendadas

- [SEC01-BP01 Separar as workloads usando contas](#)
- [SEC01-BP02 Proteger as propriedades e o usuário raiz das contas](#)

SEC01-BP01 Separar as workloads usando contas

Estabeleça barreiras de proteção e isolamento entre workloads e ambientes (como de produção, desenvolvimento e teste) por meio de uma estratégia de várias contas. A separação em nível de conta é altamente recomendável, pois ela oferece um limite de isolamento robusto para segurança, faturamento e acesso.

Resultado desejado: uma estrutura de conta que isola operações na nuvem, workloads não relacionadas e ambientes em contas separadas, aumentando a segurança em toda a infraestrutura de nuvem.

Antipadrões comuns:

- Colocação de várias workloads não relacionadas com diferentes níveis de confidencialidade na mesma conta.
- Estrutura de unidade organizacional (UO) definida de forma inadequada.

Benefícios do estabelecimento desta prática recomendada:

- Redução do escopo de impacto se uma workload for acessada acidentalmente.
- Governança central de acesso a serviços, recursos e regiões da AWS.
- Manutenção da segurança da infraestrutura de nuvem com políticas e administração centralizada de serviços de segurança.
- Criação de contas automatizada e processo de manutenção.
- Auditoria centralizada da infraestrutura de conformidade e requisitos regulatórios.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientação de implementação

As Contas da AWS oferecem um limite de isolamento de segurança entre workloads ou recursos que operam em diferentes níveis de confidencialidade. Para utilizar esse limite de isolamento, a AWS oferece ferramentas para gerenciar em grande escala suas workloads de nuvem por meio de uma estratégia de várias contas. Para ter orientações sobre os conceitos, os padrões e a implementação de uma estratégia de várias contas na AWS, consulte [Organizar seu ambiente da AWS com o uso de várias contas](#).

Quando você tem várias Contas da AWS no gerenciamento central, elas devem ser organizadas em uma hierarquia definida por camadas de unidades organizacionais (UOs). Desse modo, os controles de segurança podem ser organizados e aplicados às UOs e às contas membros, estabelecendo controles preventivos consistentes nas contas membros da organização. Os controles de segurança são herdados, permitindo que você filtre as permissões disponíveis para as contas membros localizadas em níveis inferiores de uma hierarquia de UOs. Um bom design aproveita essa herança para reduzir o número e a complexidade das políticas de segurança necessárias para obter os controles de segurança desejados para cada conta membro.

O [AWS Organizations](#) e o [AWS Control Tower](#) são dois serviços que você pode utilizar para implementar e gerenciar essa estrutura de várias contas em seu ambiente da AWS. O AWS Organizations possibilita organizar as contas em uma hierarquia definida por uma ou mais camadas

de UOs, em que cada UO contém uma série de contas membros. As [políticas de controle de serviços](#) (SCPs) permitem que o administrador da organização estabeleça controles preventivos detalhados nas contas membros, e o [AWS Config](#) pode ser utilizado para estabelecer controles proativos e de detecção nessas contas. Muitos serviços da AWS [integram-se ao AWS Organizations](#) para oferecer controles administrativos delegados e realizar tarefas específicas do serviço em todas as contas membros da organização.

Estruturado sobre o AWS Organizations, o [AWS Control Tower](#) oferece práticas recomendadas de um clique para um ambiente da AWS de várias contas com uma [zona de pouso](#). A zona de pouso é o ponto de entrada para o ambiente de várias contas estabelecido pelo Control Tower. O Control Tower oferece vários [benefícios](#) em comparação com o AWS Organizations. Três benefícios que oferecem governança aprimorada de contas são:

- Barreiras de proteção de segurança obrigatórias e integradas que são aplicadas automaticamente às contas admitidas na organização.
- Barreiras de proteção opcionais que podem ser ativadas ou desativadas em determinado conjunto de UOs.
- O [AWS Control Tower Account Factory](#) oferece implantação automatizada de contas que contêm linhas de base aprovadas e opções de configuração em sua organização.

Etapas da implementação

1. Projetar uma estrutura de unidade organizacional: uma estrutura de unidade organizacional projetada adequadamente reduz o trabalho de gerenciamento necessário para criar e manter políticas de controle de serviços e outros controles de segurança. Sua estrutura de unidade organizacional deve estar [alinhada com as necessidades, a confidencialidade dos dados e a estrutura de workload de sua empresa](#).
2. Criar uma zona de pouso para seu ambiente de várias contas: uma zona de pouso oferece uma base consistente de infraestrutura e segurança na qual sua organização pode desenvolver, executar e implantar workloads com rapidez. É possível usar uma [zona de pouso personalizada ou o AWS Control Tower](#) para orquestrar seu ambiente.
3. Estabelecer barreiras de proteção: implemente barreiras de proteção consistentes para seu ambiente por meio da zona de pouso. O AWS Control Tower oferece uma lista de controles [obrigatórios](#) e [opcionais](#) que podem ser implantados. Os controles obrigatórios são implantados automaticamente na implementação do Control Tower. Leia a lista de controles opcionais e altamente recomendados e implemente controles adequados às suas necessidades.

4. Restringir o acesso a regiões adicionadas recentemente: para novas Regiões da AWS, recursos do IAM, como usuários e perfis, serão propagados somente para as regiões especificadas. Essa ação pode ser realizada por meio do [console ao usar o Control Tower](#) ou ajustando as políticas de permissões do [IAM no AWS Organizations](#).
5. Considerar o AWS [CloudFormation StackSets](#): o StackSets ajuda você a implantar recursos, como grupos, políticas e perfis do IAM em diferentes regiões e Contas da AWS por meio de um modelo aprovado.

Recursos

Práticas recomendadas relacionadas:

- [SEC02-BP04 Contar com um provedor de identidades centralizado](#)

Documentos relacionados:

- [AWS Control Tower](#)
- [Diretrizes de auditoria de segurança da AWS](#)
- [Práticas recomendadas do IAM](#)
- [Usar o CloudFormation StackSets para fornecer recursos entre várias regiões e Contas da AWS](#)
- [Perguntas frequentes sobre o Organizations](#)
- [Terminologia e conceitos do AWS Organizations](#)
- [Práticas recomendadas para políticas de controle de serviços do AWS Organizations em um ambiente de várias contas](#)
- [Guia de referência de gerenciamento de contas da AWS](#)
- [Organização do ambiente usando várias contas da AWS](#)

Vídeos relacionados:

- [Habilitar a adoção da AWS em grande escala com automação e governança](#)
- [Práticas recomendadas de segurança de acordo com o Well-Architected](#)
- [Criar e administrar várias contas com o uso do AWS Control Tower](#)
- [Habilitar o Control Tower para organizações existentes](#)

Workshops relacionados:

- [Dia de imersão no Control Tower](#)

SEC01-BP02 Proteger as propriedades e o usuário raiz das contas

O usuário raiz é o mais privilegiado de uma Conta da AWS, com acesso administrativo integral a todos os recursos da conta, e em alguns casos não pode ser restringido por políticas de segurança. Desabilitar o acesso programático ao usuário raiz, estabelecer controles apropriados para ele e evitar o uso rotineiro desse usuário ajuda a reduzir o risco de exposição acidental das credenciais raiz e o subsequente comprometimento do ambiente de nuvem.

Resultado desejado: proteger o usuário raiz ajuda a reduzir a chance de danos acidentais ou intencionais decorrentes do mau uso das respectivas credenciais. Estabelecer controles de detecção também pode alertar o pessoal apropriado quando se realizam ações utilizando o usuário raiz.

Antipadrões comuns:

- Utilizar o usuário raiz para outras tarefas que não sejam aquelas que exigem credenciais do usuário raiz.
- Negligenciar os testes dos planos de contingência regularmente a fim de verificar a funcionalidade da infraestrutura, dos processos e dos funcionários essenciais durante uma emergência.
- Considerar apenas o fluxo típico de login de contas e não considerar nem testar métodos de recuperação de contas alternativos.
- Não lidar com DNS, servidores de e-mail e operadoras de telefonia como parte do perímetro de segurança essencial, pois eles são usados no fluxo de recuperação de contas.

Benefícios do estabelecimento desta prática recomendada: proteger o acesso ao usuário raiz cria a confiança de que as ações em sua conta estão controladas e auditadas.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientação de implementação

A AWS oferece muitas ferramentas para ajudar a proteger sua conta. No entanto, como algumas dessas medidas não estão habilitadas por padrão, é necessário implementá-las diretamente. Leve em consideração essas recomendações como etapas fundamentais para proteger sua Conta da

AWS. Ao implementar essas etapas, é importante criar um processo para avaliar e monitorar os controles de segurança de forma contínua.

Ao criar uma Conta da AWS pela primeira vez, você começa com uma identidade que tem acesso completo a todos os recursos e serviços da AWS na conta. Essa identidade é chamada de usuário raiz da Conta da AWS. É possível fazer login como usuário raiz usando o endereço de e-mail e a senha utilizados para criar a conta. Devido ao acesso elevado concedido ao usuário raiz da AWS, é necessário limitar o uso do usuário raiz da AWS à realização de tarefas que [o exigam especificamente](#). As credenciais de login do usuário raiz devem ser bem protegidas, e a autenticação multifator (MFA) sempre deve ser habilitada para o usuário raiz da Conta da AWS.

Além do fluxo de autenticação normal para fazer login com seu usuário raiz usando um nome de usuário, senha e o dispositivo de autenticação multifator (MFA), há fluxos de recuperação de contas para fazer login com seu usuário raiz da Conta da AWS com o endereço de e-mail e o número de telefone associados à sua conta. Dessa forma, é igualmente importante proteger a conta de e-mail do usuário raiz para a qual o e-mail de recuperação é enviado e o número de telefone associado à conta. Além disso, considere possíveis dependências circulares em que o endereço de e-mail associado ao usuário raiz é hospedado em servidores de e-mail ou recursos de serviço de nome de domínio (DNS) da mesma Conta da AWS.

Ao usar o AWS Organizations, há várias Contas da AWS, e cada uma tem um usuário raiz. Uma conta é designada como a conta de gerenciamento e várias camadas de contas membros podem ser adicionadas à conta de gerenciamento. Priorize a proteção do usuário raiz de sua conta de gerenciamento e, depois, os usuários raiz das contas membros. A estratégia para proteger o usuário raiz de sua conta de gerenciamento pode diferir da utilizada nos usuários raiz de suas contas membros, e é possível implementar controles de segurança preventivos nos usuários raiz dessas contas.

Etapas da implementação

As etapas de implementação a seguir são recomendadas para estabelecer controles para o usuário raiz. Quando aplicável, as recomendações têm referências cruzadas com o [Benchmark do CIS AWS Foundations versão 1.4.0](#). Além dessas etapas, consulte as [Diretrizes de práticas recomendadas da AWS](#) para proteger os recursos e a Conta da AWS.

Controles preventivos

1. Configure [informações de contato](#) precisos para a conta.

- a. Essas informações são usadas para o fluxo de recuperação de senha perdida, o fluxo de recuperação de conta de dispositivo MFA perdida e para comunicações com sua equipe sobre segurança crítica.
 - b. Utilize um endereço de e-mail hospedado por seu domínio corporativo, preferencialmente uma lista de distribuição, como o endereço de e-mail do usuário raiz. O uso de uma lista de distribuição em vez da conta de e-mail de um indivíduo oferece redundância e continuidade adicionais para o acesso à conta raiz por longos períodos.
 - c. O número de telefone listado nas informações de contato deve ser um telefone dedicado e seguro para esse fim. O número de telefone não deve ser listado nem compartilhado com ninguém.
2. Não crie chaves de acesso para o usuário raiz. Se houver chaves de acesso, remova-as (CIS 1.4).
- a. Elimine todas as credenciais programáticas de longa duração (chaves de acesso e secretas) para o usuário raiz.
 - b. Se já houver chaves de acesso do usuário raiz, será necessário fazer a transição dos processos que utilizam essas chaves para utilizar chaves de acesso temporárias de um perfil do AWS Identity and Access Management (IAM), depois [excluir as chaves de acesso do usuário raiz](#).
3. Determine se você precisa armazenar credenciais para o usuário raiz.
- a. Ao usar o AWS Organizations para criar contas membros, a senha inicial do usuário raiz em novas contas membros é definida como um valor aleatório que não é exposto a você. Se necessário, considere utilizar o fluxo de redefinição de senha de sua conta de gerenciamento do AWS Organizations para [obter acesso à conta membro](#).
 - b. Para Contas da AWS autônomas ou a conta de gerenciamento do AWS Organizations, considere criar e armazenar de forma segura as credenciais do usuário raiz. Ativar a MFA para o usuário raiz
4. Ative os controles preventivos para os usuários raiz das contas membros em ambientes de várias contas da AWS.
- a. Considere habilitar a barreira de proteção preventiva [Desautorizar criação de chaves de acesso raiz para o usuário raiz](#) para contas membros.
 - b. Considere habilitar a barreira de proteção preventiva [Desautorizar criação como um usuário raiz](#) para contas membros.
5. Se você precisar de credenciais para o usuário raiz:
- a. Use uma senha complexa.

- b. Ative a autenticação multifator (MFA) para o usuário raiz, especialmente para contas (pagantes) de gerenciamento do AWS Organizations (CIS 1.5).
 - c. Considere o uso de dispositivos de MFA de hardware para ter resiliência e segurança, pois os dispositivos de uso único reduzem as chances de os dispositivos que contêm seus códigos de MFA serem reutilizados para outros fins. Garanta que os dispositivos de MFA de hardware alimentados por bateria sejam substituídos regularmente. (CIS 1.6)
 - Para configurar a MFA para o usuário raiz, siga as instruções para habilitar uma [MFA virtual](#) ou um [dispositivo de MFA de hardware](#).
 - d. Considere registrar vários dispositivos de MFA para backup. [Até oito dispositivos de MFA são permitidos por conta](#).
 - Registrar mais de um dispositivo de MFA para o usuário raiz desabilita automaticamente o [fluxo para recuperar sua conta se o dispositivo de MFA for perdido](#).
 - e. Armazene a senha com segurança e considere as dependências circulares se for armazenar a senha eletronicamente. Não armazene a senha de uma forma que exija o acesso à mesma Conta da AWS para obtê-la.
6. Opcional: considere estabelecer um cronograma de alternância de senha periódica para o usuário raiz.
- As práticas recomendadas de gerenciamento de credenciais dependem de seus requisitos regulatórios e de política. Os usuários raiz protegidos por MFA não dependem da senha como um único fator de autenticação.
 - [A alteração periódica da senha de usuário raiz](#) reduz o risco de mau uso de uma senha exposta acidentalmente.

Controles de detecção

- Crie alarmes para detectar o uso das credenciais raiz (CIS 1.7). [Quando habilitado, o Amazon GuardDuty](#) monitora e alerta o uso de credenciais da API do usuário raiz por meio da descoberta [RootCredentialUsage](#).
- Avalie e implemente os controles de detecção incluídos no [pacote de conformidade do Pilar de segurança do AWS Well-Architected para AWS Config](#) ou, se usar o AWS Control Tower, os [controles altamente recomendados](#) disponíveis no Control Tower.

Orientação operacional

- Determine quem na organização deve ter acesso às credenciais do usuário raiz.
 - Use uma regra de duas pessoas de forma que um indivíduo tenha acesso a todas as credenciais necessárias e MFA para obter acesso de usuário raiz.
 - Verifique se é a organização, e não um único indivíduo, que mantém controle sobre o número de telefone e alias de e-mail associados à conta (que são utilizados para redefinição de senha e fluxo de redefinição de MFA).
- Utilize o usuário raiz apenas como uma exceção (CIS 1.7).
 - O usuário raiz da AWS não deve ser usado para tarefas diárias, mesmo que sejam tarefas administrativas. Somente faça login como usuário raiz para realizar [tarefas da AWS que o exigem especificamente](#). Todas as outras ações devem ser realizadas por outros usuários que assumem perfis apropriados.
- Confira periodicamente se o acesso ao usuário raiz está funcionando de forma que os procedimentos sejam testados antes de uma situação de emergência que exija o uso das credenciais do usuário raiz.
- Confira periodicamente se o endereço de e-mail associado à conta e os listados em [Contatos alternativos](#) funcionam. Monitore as caixas de entrada de e-mail das quais você recebe notificações de segurança <abuse@amazon.com>. Além disso, garanta que todos os números de telefone associados à conta estejam funcionando.
- Prepare um procedimento de resposta a incidentes para responder ao mau uso da conta raiz. Consulte o [Guia de resposta a incidentes de segurança da AWS](#) e as práticas recomendadas na [seção “Resposta a incidentes” do whitepaper Pilar Segurança](#) para ter mais informações sobre como criar uma estratégia de resposta a incidentes para sua Conta da AWS.

Recursos

Práticas recomendadas relacionadas:

- [SEC01-BP01 Separar as workloads usando contas](#)
- [SEC02-BP01 Usar mecanismos de login fortes](#)
- [SEC03-BP02 Conceder acesso com privilégio mínimo](#)
- [SEC03-BP03 Estabelecer processo de acesso de emergência](#)
- [SEC10-BP05 Acesso pré-provisionado](#)

Documentos relacionados:

- [AWS Control Tower](#)
- [Diretrizes de auditoria de segurança da AWS](#)
- [Práticas recomendadas do IAM](#)
- [Amazon GuardDuty: alerta de uso de credenciais raiz](#)
- [Orientações passo a passo sobre como monitorar o uso de credenciais raiz por meio do CloudTrail](#)
- [Tokens de MFA aprovados para uso com a AWS](#)
- Implementar [o acesso de quebra de vidro](#) na AWS
- [Os dez principais itens de segurança para aprimorar sua Conta da AWS](#)
- [O que fazer se eu notar atividade não autorizada em minha Conta da AWS?](#)

Vídeos relacionados:

- [Habilitar a adoção da AWS em grande escala com automação e governança](#)
- [Práticas recomendadas de segurança de acordo com o Well-Architected](#)
- [Limitar o uso de credenciais raiz da AWS](#) do AWS re:inforce 2022: Práticas recomendadas de segurança com o AWS IAM

Exemplos e laboratórios relacionados:

- [Laboratório: Conta da AWS e usuário raiz](#)

Operar workloads com segurança

Operar workloads com segurança abrange todo o ciclo de vida de uma workload, desde o design até a criação, execução e melhoria contínua. Uma das formas de melhorar a capacidade de operar com segurança na nuvem é adotar uma abordagem organizacional de governança. Governança significa orientar as decisões de forma consistente, sem depender apenas do bom senso das pessoas envolvidas. O modelo e o processo de governança são a forma como você responde à pergunta “Como sei que os objetivos de controle para determinada workload foram atendidos e são apropriados para essa workload?”. Ter uma abordagem consistente para tomar decisões acelera a implantação de workloads e ajuda a elevar o nível do recurso de segurança em sua organização.

Para operar sua carga de trabalho com segurança, você deve aplicar as melhores práticas gerais a todas as áreas de segurança. Use os requisitos e os processos que você definiu em excelência operacional em nível de carga de trabalho e também organizacional e aplique-os a todas as áreas. Manter-se atualizado com as recomendações da AWS e do setor e a inteligência de ameaças ajuda você a desenvolver seu modelo de ameaças e objetivos de controle. A automação de processos, testes e validação de segurança ajuda a escalar suas operações de segurança.

A automação possibilita consistência e repetibilidade dos processos. As pessoas podem ser boas em várias coisas, mas fazer a mesma coisa repetidamente e de forma consistente sem errar não é uma delas. Mesmo com runbooks bem escritos, você corre o risco de que as pessoas não realizem tarefas repetitivas de forma consistente. Isso acontece quando as pessoas têm diferentes responsabilidades e precisam responder a alertas desconhecidos. A automação, no entanto, responde sempre da mesma forma. A melhor maneira de implantar aplicações é por meio da automação. O código que executa a implantação pode ser testado e usado para realizar a implantação. Isso aumenta a confiança no processo de mudança e reduz o risco de uma mudança com falha.

Para verificar se a configuração atende aos seus objetivos de controle, primeiro teste a automação e a aplicação implantada em um ambiente de não produção. Dessa forma, você pode testar a automação para comprovar que ela executou todas as etapas corretamente. Você também receberá feedback antecipado no ciclo de desenvolvimento e implantação, reduzindo a possibilidade de refazer o trabalho. Para reduzir a probabilidade de erros de implantação, faça alterações de configuração por código, não por pessoa. Caso precise reimplantar uma aplicação, a automação facilitará muito. Conforme se definem objetivos de controle adicionais, é possível adicioná-los facilmente à automação para todas as workloads.

Em vez de ter proprietários de workloads individuais investindo em segurança específica para elas, economize tempo usando recursos comuns e componentes compartilhados. Alguns exemplos de serviços que várias equipes podem consumir incluem o processo de criação de conta da AWS, identidade centralizada para pessoas, configuração de log comum e criação de imagem de base de contêiner e AMI. Essa abordagem pode ajudar os criadores a melhorar os tempos de ciclo da workload e atender consistentemente aos objetivos de controle de segurança. Quando as equipes são mais consistentes, você pode validar os objetivos de controle e relatar melhor a postura de controle e posição de risco para as partes interessadas.

Práticas recomendadas

- [SEC01-BP03 Identificar e validar objetivos de controle](#)
- [SEC01-BP04 Manter-se a par das ameaças e recomendações de segurança](#)

- [SEC01-BP05 Reduzir o escopo do gerenciamento de segurança](#)
- [SEC01-BP06 Automatizar a implantação de controles de segurança padrão](#)
- [SEC01-BP07 Identificar ameaças e priorizar mitigações com o uso de um modelo de ameaça](#)
- [SEC01-BP08 Avaliar e implementar regularmente novos serviços e recursos de segurança](#)

SEC01-BP03 Identificar e validar objetivos de controle

Com base em seus requisitos de conformidade e riscos identificados no modelo de ameaça, derive e valide os objetivos de controle e os controles que você precisa aplicar à workload. A validação contínua de objetivos de controle e controles ajuda a medir a eficácia da mitigação de riscos.

Resultado desejado: os objetivos de controle de segurança da sua empresa estão bem definidos e alinhados aos seus requisitos de conformidade. Os controles são implementados e aplicados por meio de automação e políticas, bem como continuamente avaliados quanto à respectiva eficácia para alcançar seus objetivos. As evidências de eficácia em determinado momento e durante um período de tempo podem ser facilmente relatadas aos auditores.

Antipadrões comuns:

- Os requisitos regulatórios, as expectativas de mercado e os padrões do setor de garantia de segurança não são claros para sua empresa.
- Seus frameworks de segurança cibernética e seus objetivos de controle estão desalinhados com relação aos requisitos de sua empresa.
- A implementação de controles não se alinha de maneira consistente e mensurável com os seus objetivos de controle.
- Você não usa a automação para relatar a eficácia de seus controles.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

Há muitos frameworks comuns de segurança cibernética que podem servir de base para seus objetivos de controle de segurança. Considere os requisitos regulatórios, as expectativas de mercado e os padrões do setor aplicáveis à sua empresa a fim de determinar quais frameworks atendem melhor às suas necessidades. Exemplos incluem [AICPA SOC 2](#), [HITRUST](#), [PCI-DSS](#), [ISO 27001](#) e [NIST SP 800-53](#).

Com relação aos objetivos de controle identificados, entenda como os serviços da AWS que você consome ajudam a atingi-los. Use o [AWS Artifact](#) para encontrar documentos e relatórios alinhados aos frameworks pretendidos que descrevam o escopo de responsabilidade atendido pela AWS e orientações sobre o escopo restante sob sua responsabilidade. Para obter mais orientações específicas aos serviços, relativas ao alinhamento com diferentes declarações de controle de framework, consulte [AWS Customer Compliance Guides](#).

Ao definir os controles que viabilizam seus objetivos, codifique a imposição usando controles preventivos e automatize a mitigação usando controles de detecção. Ajude a evitar configurações e ações de recursos irregulares no AWS Organizations usando [políticas de controle de serviços \(SCPs\)](#). Implemente regras no [AWS Config](#) para monitorar e relatar recursos irregulares e, em seguida, mude-as para um modelo de imposição quando acreditar que elas terão o comportamento desejado. Para implantar conjuntos de regras predefinidas e gerenciadas que se alinham aos seus frameworks de segurança cibernética, avalie o uso de [padrões do AWS Security Hub](#) como sua primeira opção. O padrão de Práticas de Segurança Básica da AWS (FSBP) e o CIS AWS Foundations Benchmark são bons pontos de partida e têm controles que se alinham a muitos objetivos que são compartilhados em vários frameworks padrão. Quando o Security Hub não tiver intrinsecamente as detecções de controle desejadas, elas poderão ser complementadas usando os [pacotes de conformidade do AWS Config](#).

Use os [pacotes de parceiros da APN](#) recomendados pela equipe do programa Aceleração da Segurança e Conformidade Global (GSCA) da AWS para obter assistência de consultores de segurança, agências de consultoria, sistemas de coleta de evidências e geração de relatórios, auditores e outros serviços complementares quando necessário.

Etapas da implementação

1. Avalie frameworks comuns de segurança cibernética e alinhe seus objetivos de controle aos escolhidos.
2. Obtenha documentação relevante sobre orientações e responsabilidades pelo uso de seu framework usando o AWS Artifact. Entenda quais partes da conformidade enquadram-se no Modelo de Responsabilidade Compartilhada da AWS e quais partes são de sua responsabilidade.
3. Use SCPs, políticas de recursos, políticas de confiança de perfil e outras barreiras de proteção para evitar configurações e ações de recursos irregulares.
4. Avalie a implantação de padrões do Security Hub e de pacotes de conformidade do AWS Config que se alinhem aos seus objetivos de controle.

Recursos

Práticas recomendadas relacionadas:

- [SEC03-BP01 Definir requisitos de acesso](#)
- [SEC04-BP01 Configurar registro em log de serviço e aplicação](#)
- [SEC07-BP01 Compreender seu esquema de classificação de dados](#)
- [OPS01-BP03 Avaliar os requisitos de governança](#)
- [OPS01-BP04 Avaliar os requisitos de conformidade](#)
- [PERF01-BP05 Usar políticas e arquiteturas de referência](#)
- [COST02-BP01 Desenvolver políticas com base nos requisitos da sua organização](#)

Documentos relacionados:

- [AWS Customer Compliance Guides](#)

Ferramentas relacionadas:

- [AWS Artifact](#)

SEC01-BP04 Manter-se a par das ameaças e recomendações de segurança

Mantenha-se a par das ameaças e mitigações mais recentes monitorando as publicações de inteligência contra ameaças do setor e os feeds de dados para atualizações. Avalie as ofertas de serviços gerenciados que são atualizadas automaticamente com base nos dados de ameaças mais recentes.

Resultado desejado: você se mantém a par das publicações do setor à medida que elas são atualizadas com as ameaças e recomendações mais recentes. Você usa a automação para detectar possíveis vulnerabilidades e exposições à medida que identifica novas ameaças. Você toma medidas de mitigação contra essas ameaças. Você adota serviços da AWS que são atualizados automaticamente com a inteligência de ameaças mais recente.

Antipadrões comuns:

- Não ter um mecanismo confiável e repetível para se manter a par das últimas informações sobre ameaças.
- Manter um inventário manual do portfólio de tecnologia, das workloads e das dependências que exigem análise humana de possíveis vulnerabilidades e exposições.
- Não ter mecanismos para atualizar workloads e dependências para as versões mais recentes disponíveis que ofereçam mitigações de ameaças conhecidas.

Benefícios do estabelecimento desta prática recomendada: usar fontes de inteligência contra ameaças para se manter em dia reduz o risco de você ignorar mudanças importantes no cenário de ameaças que podem afetar seus negócios. Ter a automação implementada para verificar, detectar e corrigir possíveis vulnerabilidades ou exposições em workloads e dependências pode ajudar você a reduzir os riscos de forma rápida e previsível, em comparação com as alternativas manuais. Isso ajuda a controlar o tempo e os custos relacionados à mitigação de vulnerabilidades.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

Analise publicações confiáveis de inteligência contra ameaças para ficar por dentro do cenário de ameaças. Consulte a base de conhecimento [MITRE ATT&CK](#) para obter documentação sobre táticas, técnicas e procedimentos (TTPs) de adversários conhecidos. Analise a lista de [Common Vulnerabilities and Exposures](#) (CVE) da MITRE para se manter a par das vulnerabilidades conhecidas em produtos nos quais você confia. Entenda os riscos críticos das aplicações web por meio do conhecido projeto [OWASP Top 10](#) do Open Worldwide Application Security Project (OWASP).

Mantenha-se a par dos eventos de segurança e das etapas de correção recomendadas da AWS por meio dos [Boletins de segurança](#) da AWS para CVEs.

Para reduzir o esforço geral e as despesas indiretas para se manter em dia, considere usar serviços da AWS que incorporam automaticamente novas informações sobre ameaças ao longo do tempo. Por exemplo, o [Amazon GuardDuty](#) mantém-se atualizado com a inteligência contra ameaças do setor para detectar comportamentos anômalos e assinaturas de ameaças em suas contas. O [Amazon Inspector](#) atualiza automaticamente o banco de dados de CVEs utilizado para seus recursos de verificação contínua. Tanto o [AWS WAF](#) quanto o [AWS Shield Advanced](#) fornecem grupos de regras gerenciadas que passam por atualizações automáticas à medida que surgem novas ameaças.

Analise o [Pilar de excelência operacional do Well-Architected](#) para gerenciamento e correção automatizados de frotas.

Etapas da implementação

- Assine atualizações de publicações de inteligência contra ameaças que sejam relevantes para sua empresa e setor. Assine os Boletins de segurança da AWS.
- Considere a adoção de serviços que incorporem automaticamente novas informações sobre ameaças, como o Amazon GuardDuty e o Amazon Inspector.
- Implemente uma estratégia de gerenciamento e correção de frotas que se alinhe às práticas recomendadas do pilar de excelência operacional do Well-Architected.

Recursos

Práticas recomendadas relacionadas:

- [SEC01-BP07 Identificar ameaças e priorizar mitigações com o uso de um modelo de ameaça](#)
- [OPS01-BP05 Avaliar o cenário de ameaças](#)
- [OPS11-BP01 Ter um processo para a melhoria contínua](#)

SEC01-BP05 Reduzir o escopo do gerenciamento de segurança

Determine se você pode reduzir seu escopo de segurança usando serviços da AWS que transferem o gerenciamento de determinados controles para serviços gerenciados da AWS. Esses serviços podem ajudar a reduzir suas tarefas de manutenção de segurança, como provisionamento de infraestrutura, configuração de software, aplicação de patches ou backups.

Resultado desejado: o escopo do gerenciamento de segurança é considerado ao selecionar serviços da AWS para a workload. O custo referente a despesas gerais de gerenciamento e a tarefas de manutenção (o custo total de propriedade ou TCO) é ponderado em relação ao custo dos serviços que você seleciona, além de outras considerações do Well-Architected. Você incorpora a documentação de controle e conformidade da AWS em seus procedimentos de avaliação e verificação de controle.

Antipadrões comuns:

- Implantar workloads sem entender completamente o Modelo de Responsabilidade Compartilhada referente aos serviços que você seleciona.

- Hospedar bancos de dados e outras tecnologias em máquinas virtuais sem ter avaliado um serviço gerenciado equivalente.
- Não incluir tarefas de gerenciamento de segurança no custo total de propriedade de tecnologias de hospedagem em máquinas virtuais em comparação com as opções de serviços gerenciados.

Benefícios do estabelecimento desta prática recomendada: o uso de serviços gerenciados pode reduzir a sobrecarga geral de gerenciar controles de segurança operacional, o que, por sua vez, diminui os riscos de segurança e o custo total de propriedade. O tempo que de outra forma seria gasto em determinadas tarefas de segurança pode ser reinvestido em tarefas que agregam maior valor aos negócios. Os serviços gerenciados também podem reduzir o escopo dos requisitos de conformidade ao transferir alguns requisitos de controle para a AWS.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Há várias maneiras de integrar os componentes da workload na AWS. A instalação e execução de tecnologias em instâncias do Amazon EC2 geralmente exigem que você assuma a maior parte da responsabilidade geral pela segurança. Para ajudar a diminuir a sobrecarga de operar determinados controles, identifique serviços gerenciados da AWS que reduzam o escopo da sua parte do Modelo de Responsabilidade Compartilhada e entenda como é possível usá-los em sua arquitetura atual. Exemplos incluem usar o [Amazon Relational Database Service \(Amazon RDS\)](#) para implantar bancos de dados, o [Amazon Elastic Kubernetes Service \(Amazon EKS\)](#) ou o [Amazon Elastic Container Service \(Amazon ECS\)](#) para orquestrar contêineres ou [opções sem servidor](#). Ao criar novas aplicações, pense em quais serviços podem ajudar a reduzir o tempo e o custo referentes à implementação e ao gerenciamento de controles de segurança.

Os requisitos de conformidade também podem ser um fator na seleção de serviços. Os serviços gerenciados podem mudar a conformidade de alguns requisitos relacionados à AWS. Converse com sua equipe de conformidade sobre quanto ela se sente confortável para auditar os aspectos dos serviços que você opera e gerencia e aceitar declarações de controle em relatórios de auditoria relevantes da AWS. Você pode fornecer os artefatos de auditoria encontrados no [AWS Artifact](#) aos seus auditores ou reguladores como evidência dos controles de segurança da AWS. Também é possível usar as orientações sobre responsabilidade fornecidas por alguns dos artefatos de auditoria da AWS para projetar sua arquitetura, bem como os [AWS Customer Compliance Guides](#). Essas orientações ajudam a determinar os controles de segurança adicionais que você deve implementar para atender aos casos de uso específicos do seu sistema.

Ao usar serviços gerenciados, familiarize-se com o processo de atualização de recursos para versões mais recentes (por exemplo, atualizar a versão de um banco de dados gerenciado pelo Amazon RDS ou o runtime de uma linguagem de programação para uma função do AWS Lambda). Embora o serviço gerenciado possa realizar essa operação para você, configurar o momento da atualização e entender o impacto em suas operações continua sendo sua responsabilidade. Ferramentas como o [AWS Health](#) podem ajudar você a monitorar e gerenciar essas atualizações em todos os ambientes.

Etapas da implementação

1. Avalie os componentes da workload que podem ser substituídos por um serviço gerenciado.
 - a. Se você estiver migrando uma workload para a AWS, considere a redução do gerenciamento (tempo e despesas) e a diminuição do risco ao avaliar se deve redefinir a hospedagem, refatorar, redefinir a plataforma, reformular, recompilar ou substituir a workload. Às vezes, investimentos adicionais no início de uma migração podem gerar economias significativas no longo prazo.
2. Pense em implementar serviços gerenciados (por ex., o Amazon RDS) em vez de instalar e gerenciar suas próprias implantações de tecnologia.
3. Use as orientações sobre responsabilidade no AWS Artifact para ajudar a determinar os controles de segurança que você deve implementar para a workload.
4. Mantenha um inventário dos recursos em uso e esteja sempre a par de novos serviços e abordagens a fim de identificar novas oportunidades para reduzir o escopo.

Recursos

Práticas recomendadas relacionadas:

- [PERF02-BP01 Selecionar as melhores opções de computação para as workloads](#)
- [PERF03-BP01 Usar um armazenamento de dados específico que melhor atenda aos seus requisitos de acesso e armazenamento de dados](#)
- [SUS05-BP03 Usar serviços gerenciados](#)

Documentos relacionados:

- [Planned lifecycle events for AWS Health](#)

Ferramentas relacionadas:

- [AWS Health](#)
- [AWS Artifact](#)
- [AWS Customer Compliance Guides](#)

Vídeos relacionados:

- [How do I migrate to an Amazon RDS or Aurora MySQL DB instance using AWS DMS?](#)
- [AWS re:Invent 2023 - Manage resource lifecycle events at scale with AWS Health](#)

SEC01-BP06 Automatizar a implantação de controles de segurança padrão

Aplique práticas modernas de DevOps ao desenvolver e implantar controles de segurança que são padrão em seus ambientes da AWS. Defina controles e configurações de segurança padrão usando modelos de infraestrutura como código (IaC), capture alterações em um sistema de controle de versão, teste as alterações como parte de um pipeline de CI/CD e automatize a implantação de mudanças em seus ambientes da AWS.

Resultado desejado: os modelos de IaC capturam controles de segurança padronizados e os submetem a um sistema de controle de versão. Os pipelines de CI/CD estão em locais que detectam mudanças e automatizam os testes e a implantação de seus ambientes da AWS. Existem barreiras de proteção para detectar e emitir alertas sobre configurações incorretas nos modelos antes de prosseguir com a implantação. As workloads são implantadas em ambientes em que há controles padrão em vigor. As equipes têm acesso para implantar configurações de serviço aprovadas por meio de um mecanismo de autoatendimento. Existem estratégias seguras de backup e recuperação para controlar configurações, scripts e dados relacionados.

Antipadrões comuns:

- Fazer alterações manuais nos controles de segurança padrão por meio de um console da web ou uma interface de linha de comandos.
- Contar com equipes de workload individuais para implementar manualmente os controles definidos por uma equipe central.
- Contar com uma equipe central de segurança para implantar controles em nível de workload a pedido de uma equipe de workload.

- Permitir que as mesmas pessoas ou equipes desenvolvam, testem e implantem scripts de automação de controle de segurança sem a separação adequada de deveres ou freios e contrapesos.

Benefícios do estabelecimento desta prática recomendada: o uso de modelos para definir controles de segurança padrão permite que você acompanhe e compare as alterações ao longo do tempo usando um sistema de controle de versão. Usar a automação para testar e implantar alterações gera padronização e previsibilidade, aumentando as chances de uma implantação bem-sucedida e reduzindo as tarefas manuais repetitivas. Fornecer um mecanismo de autoatendimento para as equipes de workload implantarem serviços e configurações aprovados reduz o risco de configuração incorreta e uso indevido. Isso também ajuda as equipes a incorporar controles logo no início no processo de desenvolvimento.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Ao seguir as práticas descritas em [SEC01-BP01 Separar as workloads usando contas](#), você terá várias Contas da AWS para diferentes ambientes gerenciados por meio do AWS Organizations. Embora cada um desses ambientes e workloads possam precisar de controles de segurança distintos, você pode padronizar alguns deles em toda a sua organização. Isso inclui integração de provedores de identidades centralizados, definição de redes e firewalls e configuração de locais padrão para armazenar e analisar logs. Assim como é possível usar a infraestrutura como código (IaC) para aplicar o mesmo rigor do desenvolvimento de código de aplicações ao provisionamento de infraestrutura, é possível usá-la para definir e implantar controles de segurança padrão.

Sempre que possível, defina controles de segurança de forma declarativa, como no [AWS CloudFormation](#), e armazene-os em um sistema de controle de origem. Use práticas de DevOps para automatizar a implantação dos controles para versões mais previsíveis, bem como testes automatizados usando ferramentas como o [AWS CloudFormation Guard](#), e detecte diferenças entre os controles implantados e a configuração desejada. Você pode usar determinados serviços, como o [AWS CodePipeline](#), o [AWS CodeBuild](#) e o [AWS CodeDeploy](#), para construir um pipeline de CI/CD. Considere as orientações fornecidas em [Organizing Your AWS Environment Using Multiple Accounts](#) para configurar esses serviços em contas próprias separadas de outros pipelines de implantação.

Você também pode definir modelos para padronizar a definição e a implantação de serviços, configurações e Contas da AWS. Essa técnica permite que uma equipe central de segurança gerencie essas definições e as forneça às equipes de workload por meio de uma abordagem de

autoatendimento. Uma maneira de fazer isso é usar o [Service Catalog](#), no qual você pode publicar modelos como produtos, os quais as equipes de workload podem incorporar em suas próprias implantações de pipeline. Se estiver usando o [AWS Control Tower](#), alguns modelos e controles estão disponíveis como ponto de partida. O Control Tower também fornece o recurso [Account Factory](#), permitindo que as equipes de workload criem Contas da AWS usando padrões definidos por você. Com esse recurso, não é preciso depender de uma equipe central para aprovar e criar contas quando elas são identificadas como necessárias pelas equipes de workload. Talvez essas contas precisem isolar diferentes componentes da workload com base em motivos como a função que eles desempenham, a confidencialidade dos dados que estão sendo processados ou o comportamento desses componentes.

Etapas da implementação

1. Determine como você armazenará e manterá os modelos em um sistema de controle de versão.
2. Crie pipelines de CI/CD para testar e implantar modelos. Defina testes para verificar se há configurações incorretas e se os modelos estão de acordo com os padrões da sua empresa.
3. Crie um catálogo de modelos padronizados para as equipes de workload implantarem serviços e Contas da AWS de acordo com suas necessidades.
4. Implemente estratégias seguras de backup e recuperação para suas configurações de controle, scripts e dados relacionados.

Recursos

Práticas recomendadas relacionadas:

- [OPS05-BP01 Usar o controle de versão](#)
- [OPS05-BP04 Usar sistemas de gerenciamento de compilação e de implantação](#)
- [REL08-BP05 Implantar alterações com automação](#)
- [SUS06-BP01 Adotar métodos que podem apresentar melhorias na sustentabilidade rapidamente](#)

Documentos relacionados:

- [Organizing Your AWS Environment Using Multiple Accounts](#)

Exemplos relacionados:

- [Automate account creation, and resource provisioning using Service Catalog, AWS Organizations, and AWS Lambda](#)
- [Strengthen the DevOps pipeline and protect data with AWS Secrets Manager, AWS KMS, and AWS Certificate Manager](#)

Ferramentas relacionadas:

- [AWS CloudFormation Guard](#)
- [Landing Zone Accelerator on AWS](#)

SEC01-BP07 Identificar ameaças e priorizar mitigações com o uso de um modelo de ameaça

Realize a modelagem de ameaças para identificar e manter um registro atualizado de possíveis ameaças e mitigações associadas para sua workload. Priorize suas ameaças e adapte as mitigações de controles de segurança para prevenir, detectar e responder. Revise e mantenha isso no contexto de sua workload e no cenário de segurança em evolução.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

O que é modelagem de ameaças?

“A modelagem de ameaças serve para identificar, comunicar e compreender as ameaças e as mitigações no contexto de proteção de algo de valor.” [The Open Web Application Security Project \(OWASP\) Application Threat Modeling](#)

Por que você deve ter um modelo de ameaças?

Os sistemas são complexos e se tornam cada vez mais intrincados e qualificados com o passar do tempo, oferecendo maior valor empresarial e maior satisfação e engajamento do cliente. Isso significa que as decisões de design de TI precisam considerar um número cada vez maior de casos de uso. Essa complexidade e o número de permutações de caso de uso geralmente tornam as abordagens não estruturadas ineficazes para encontrar e mitigar ameaças. Em vez disso, você precisa de uma abordagem sistemática para enumerar as possíveis ameaças ao sistema, elaborar mitigações e priorizá-las a fim de garantir que os recursos limitados de sua organização tenham impacto máximo na melhoria do procedimento geral de segurança do sistema.

A modelagem de ameaças foi projetada para oferecer essa abordagem sistemática, com o objetivo de encontrar e resolver problemas na fase inicial do processo de design, quando as mitigações têm custo e esforço relativamente baixos em comparação com a fase posterior do ciclo de vida. Essa abordagem está alinhada ao princípio de [segurança shift left](#) (mover para a esquerda) do setor. Por fim, a modelagem de ameaças é integrada ao processo de gerenciamento de riscos de uma organização e ajuda a impulsionar as decisões sobre quais controles implementar usando uma abordagem orientada a ameaças.

Quando a modelagem de ameaças deve ser realizada?

Inicie a modelagem de ameaças o quanto antes no ciclo de vida de sua workload. Isso oferece a você maior flexibilidade sobre o que fazer com as ameaças identificadas. Muito semelhante aos bugs de software, quanto mais cedo você identificar ameaças, mais econômico será resolvê-las. Um modelo de ameaças é um documento ativo e deve continuar a evoluir à medida que suas workloads mudam. Revise seus modelos de ameaça no decorrer do tempo, inclusive quando há uma alteração importante ou uma alteração no cenário de ameaças ou ao adotar um novo recurso ou serviço.

Etapas da implementação

Como podemos realizar a modelagem de ameaças?

Há muitas formas diferentes de realizar a modelagem de ameaças. Muito semelhante às linguagens de programação, há vantagens e desvantagens em cada uma, e é necessário escolher a forma mais adequada para você. Uma abordagem é começar com o [Shostack's 4 Question Frame for Threat Modeling](#) (Estrutura de quatro perguntas do Shostack para modelagem de ameaças), que apresenta perguntas abertas a fim de oferecer estrutura ao seu exercício de modelagem de ameaças:

1. Em que você está trabalhando?

A finalidade dessa pergunta é ajudar você a entender e chegar a um acordo sobre o sistema que você está construindo e os detalhes sobre ele que são relevantes para a segurança. A criação de um modelo ou um diagrama é a forma mais comum de responder a essa pergunta, pois ele ajuda você a visualizar o que você está construindo; por exemplo, usando um [fluxograma de dados](#). Escrever as suposições e os detalhes importantes sobre seu sistema também ajuda a definir o que está no escopo. Isso permite que todos que estão contribuindo para o modelo de ameaças se concentrem na mesma coisa e evitem desvios demorados para tópicos fora do escopo (inclusive versões desatualizadas do sistema). Por exemplo, se você estiver criando uma aplicação web, provavelmente não vale a pena criar uma modelagem de ameaças da sequência de inicialização

confiável do sistema operacional para clientes de navegador, pois não há nenhuma possibilidade de seu design ter influência nisso.

2. O que pode dar errado?

É nessa fase que você identifica ameaças ao seu sistema. Ameaças são ações ou eventos acidentais ou intencionais que têm impactos indesejados que podem afetar a segurança de seu sistema. Sem um claro entendimento do que pode dar errado, não há o que fazer sobre isso.

Não há uma lista canônica do que pode dar errado. A criação dessa lista exige etapas de brainstorming e colaboração entre todas as pessoas de sua equipe e as [pessoas relevantes envolvidas](#) no exercício de modelagem de ameaças. Você pode auxiliar suas etapas de brainstorming utilizando um modelo para identificar ameaças, como o [STRIDE](#), que sugere categorias diferentes para avaliar: Spoofing (Falsificação), Tampering (Violação), Repudiation (Repúdio), Information disclosure (Divulgação de informações), Denial of service (Negação de serviço) e Elevation of privilege (Elevação de privilégio). Além disso, talvez você queira auxiliar as etapas de brainstorming revisando as listas existentes e a pesquisa para inspiração, como o [OWASP Top 10](#), o [HiTrust Threat Catalog](#) e o catálogo de ameaças de sua própria organização.

3. O que estamos fazendo a respeito?

Como no caso da primeira pergunta, não há uma lista canônica de todas as mitigações possíveis. A entradas nessa etapa são as ameaças identificadas, as pessoas e as áreas de melhoria da etapa anterior.

Segurança e conformidade são [responsabilidades compartilhadas entre você e a AWS](#).

É importante entender que ao perguntar “O que vamos fazer a respeito?” você também pergunte “Quem é responsável por fazer algo a respeito?”. Entender o equilíbrio entre suas responsabilidades e as da AWS ajuda a definir o escopo de seu exercício de modelagem de ameaças para as mitigações que estão sob seu controle, que, geralmente, são uma combinação de opções de configuração de serviços da AWS e suas mitigações específicas ao sistema.

No que se refere à responsabilidade compartilhada da AWS, você descobrirá que os [serviços da AWS estão no escopo de muitos programas de conformidade](#). Esses programas ajudam você a entender os controles sólidos implementados na AWS para manter a segurança e a conformidade da nuvem. Os relatórios de auditoria desses programas estão disponíveis para download para clientes da AWS do [AWS Artifact](#).

Seja quais forem os serviços da AWS que você esteja utilizando, sempre há um elemento de responsabilidade do cliente, e as mitigações alinhadas a essas responsabilidades devem ser

incluídas em seu modelo de ameaças. Para mitigações de controle de segurança dos próprios serviços da AWS, convém considerar a implementação de controles de segurança em todos os domínios; por exemplo, domínios como gerenciamento de identidade e acesso (autenticação e autorização), proteção de dados (em repouso e em trânsito), segurança de infraestrutura, registro em log e monitoramento. A documentação de cada serviço da AWS tem um [capítulo dedicado à segurança](#) que oferece orientações sobre os controles de segurança a serem considerados como mitigações. É importante considerar o código que você está escrevendo e suas dependências e pensar nos controles que você poderia implementar para resolver essas ameaças. Esses controles podem ser fatores como [validação de entrada](#), [processamento de sessões](#) e [processamento de limites](#). Com frequência, a maioria das vulnerabilidades é introduzida em código personalizado, então concentre-se nessa área.

4. Fizemos um bom trabalho?

O objetivo é a sua equipe e a organização aprimorarem a qualidade dos modelos de ameaças e a velocidade na qual você está realizando a modelagem de ameaças no decorrer do tempo. Essas melhorias vêm de uma combinação de prática, aprendizado, instrução e revisão. Para se aprofundar e trabalhar, é recomendável que você e sua equipe concluam o [curso de treinamento Threat modeling the right way for builders](#) (Modelagem de ameaças da maneira certa para desenvolvedores) ou o respectivo [workshop](#). Além disso, se você estiver procurando orientações sobre como integrar a modelagem de ameaças ao ciclo de vida do desenvolvimento de aplicações da organização, consulte a publicação [Como abordar a modelagem de ameaças](#) no Blog de segurança da AWS.

Threat Composer

Para ajudar e fornecer orientações ao criar a modelagem de ameaças, considere usar a ferramenta [Threat Composer](#), que visa reduzir o tempo de maturação na modelagem de ameaças. Essa ferramenta ajuda a fazer o seguinte:

- Escrever declarações úteis sobre ameaças alinhadas à [gramática de ameaças](#) que funcionem em um fluxo de trabalho natural e não linear.
- Gerar um modelo de ameaça legível por humanos.
- Gerar um modelo de ameaça legível por máquina para permitir tratar os modelos de ameaças como código.
- Ajudar a identificar rapidamente as áreas de melhoria da qualidade e de cobertura usando o painel do Insights.

Para obter mais referências, acesse o Threat Composer e alterne para o Example Workspace definido pelo sistema.

Recursos

Práticas recomendadas relacionadas:

- [SEC01-BP03 Identificar e validar objetivos de controle](#)
- [SEC01-BP04 Manter-se a par das ameaças e recomendações de segurança](#)
- [SEC01-BP05 Reduzir o escopo do gerenciamento de segurança](#)
- [SEC01-BP08 Avaliar e implementar regularmente novos serviços e recursos de segurança](#)

Documentos relacionados:

- [Como abordar a modelagem de ameaças](#) (Blog de segurança da AWS)
- [NIST: Guia para modelagem de ameaças de sistemas centrados em dados](#)

Vídeos relacionados:

- [AWS Summit ANZ 2021: Como abordar a modelagem de ameaças](#)
- [AWS Summit ANZ 2022: Escalar a segurança: otimizar para ter uma entrega rápida e segura](#)

Treinamento relacionado:

- [Threat modeling the right way for builders \(Modelagem de ameaças da maneira certa para desenvolvedores\): treinamento autoguiado virtual do AWS Skill Builder](#)
- [Threat modeling the right way for builders – AWS Workshop](#) (Modelagem de ameaças da maneira certa para desenvolvedores)

Ferramentas relacionadas:

- [Threat Composer](#)

SEC01-BP08 Avaliar e implementar regularmente novos serviços e recursos de segurança

Avalie e implemente serviços e recursos de segurança da AWS e de parceiros da AWS que ajudem você a desenvolver o procedimento de segurança de suas workloads.

Resultado desejado: há uma prática padrão em vigor que informa você sobre os novos recursos e serviços lançados pela AWS e por parceiros da AWS. Você avalia como esses novos recursos influenciam o design dos controles novos e atuais de seus ambientes e workloads.

Antipadrões comuns:

- Você não assina blogs e feeds RSS da AWS para tomar conhecimento rapidamente de novos recursos e serviços relevantes.
- Você recorre a notícias e atualizações sobre serviços e recursos de segurança de fontes secundárias.
- Você não incentiva os usuários da AWS da sua organização a se manterem informados sobre as atualizações mais recentes

Benefícios do estabelecimento desta prática recomendada: quando você se mantém em dia sobre novos serviços e recursos de segurança, pode tomar decisões respaldadas sobre a implementação de controles em seus ambientes de nuvem e workloads. Essas fontes ajudam a aumentar a conscientização sobre o dinâmico cenário de segurança e como os serviços da AWS podem ser usados para impedir ameaças novas e emergentes.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: baixo

Orientações para a implementação

A AWS informa os clientes sobre novos serviços e recursos de segurança por meio de vários canais:

- [Quais as novidades da AWS?](#)
- [AWS News Blog](#)
- [AWS Security Blog](#)
- [Boletins de segurança da AWS](#)
- [Visão geral da documentação da AWS](#)

Você pode assinar o tópico de [atualizações diárias de recursos da AWS](#) usando o Amazon Simple Notification Service (Amazon SNS) para obter um resumo diário abrangente das atualizações. Alguns serviços de segurança, como o [Amazon GuardDuty](#) e o [AWS Security Hub](#), fornecem seus próprios tópicos de SNS para informar sobre novos padrões e descobertas e outras atualizações desses serviços específicos.

Novos serviços e recursos também são anunciados e descritos em detalhes durante [conferências, eventos e webinars](#) realizados anualmente ao redor do mundo. Merecem menção especial a conferência anual sobre segurança do [AWS re:Inforce](#) e a conferência mais genérica do [AWS re:Invent](#). Os canais de notícias da AWS mencionados anteriormente compartilham esses anúncios de conferência sobre segurança e outros serviços. Além disso, você pode assistir on-line a sessões temáticas educacionais aprofundadas no [canal AWS Events](#) no YouTube.

Você também pode consultar a [equipe de sua Conta da AWS](#) sobre as últimas atualizações e recomendações de serviços de segurança. Você pode entrar em contato com sua equipe por meio do [formulário de suporte de vendas](#) se não tiver as informações de contato direto da equipe. Da mesma forma, se você tiver assinado o [AWS Enterprise Support](#), receberá atualizações semanais do seu gerente técnico de contas (TAM) e poderá agendar uma reunião de revisão regular com ele.

Etapas da implementação

1. Assine vários blogs e boletins com o leitor de RSS de sua preferência ou o tópico de atualizações diárias de recursos do SNS.
2. Avalie de quais eventos da AWS você deve participar para saber em primeira mão sobre novos recursos e serviços.
3. Agende reuniões com a equipe de sua Conta da AWS para esclarecer dúvidas sobre a atualização de serviços e recursos de segurança.
4. Considere a possibilidade de assinar o Enterprise Support para consultar regularmente um gerente técnico de contas (TAM).

Recursos

Práticas recomendadas relacionadas:

- [PERF01-BP01 Conhecer e compreender os serviços e recursos de nuvem disponíveis](#)
- [COST01-BP07 Manter-se atualizado com os novos lançamentos de serviços](#)

Gerenciamento de identidade e acesso

Para usar os serviços da AWS, você deve conceder aos usuários e aplicações acesso a recursos em suas contas da AWS. À medida que executa mais workloads na AWS, você precisa de um gerenciamento de identidade robusto e de permissões implementadas para garantir que as pessoas certas tenham acesso aos recursos certos nas condições certas. A AWS oferece uma grande variedade de recursos para ajudar a gerenciar identidades humanas e de máquinas e suas permissões. As práticas recomendadas para esses recursos se encaixam em duas áreas principais.

Tópicos

- [Gerenciamento de identidades](#)
- [Gerenciamento de permissões](#)

Gerenciamento de identidades

Há dois tipos de identidade que você precisa gerenciar para operar workloads seguras da AWS.

- Identidades humanas: administradores, desenvolvedores, operadores e clientes de suas aplicações precisam de uma identidade para acessar seus ambientes e aplicações da AWS. Eles podem ser membros da sua organização ou usuários externos com quem você colabora e que interagem com seus recursos da AWS por meio de um navegador da web, uma aplicação cliente, um aplicativo móvel ou de ferramentas interativas de linha de comando.
- Identidades de máquina: aplicações de workload, ferramentas operacionais e componentes precisam de uma identidade para solicitar serviços da AWS, como ler dados. Essas identidades incluem máquinas em execução em seu ambiente da AWS, como instâncias do Amazon EC2 ou funções do AWS Lambda. Você também pode gerenciar identidades de máquina para partes externas que precisam de acesso. Além disso, você pode ter máquinas fora da AWS que precisam de acesso ao seu ambiente da AWS.

Práticas recomendadas

- [SEC02-BP01 Usar mecanismos de login fortes](#)
- [SEC02-BP02 Usar credenciais temporárias](#)
- [SEC02-BP03 Armazenar e usar segredos com segurança](#)
- [SEC02-BP04 Contar com um provedor de identidades centralizado](#)

- [SEC02-BP05 Fazer a auditoria e a alternância periódica das credenciais](#)
- [SEC02-BP06 Utilizar grupos de usuários e atributos](#)

SEC02-BP01 Usar mecanismos de login fortes

Os logins (autenticação com credenciais de login) podem apresentar riscos quando não são usados mecanismos, como autenticação multifator (MFA), especialmente em situações em que as credenciais de login foram divulgadas acidentalmente ou podem ser deduzidas com facilidade. Utilize mecanismos de login fortes para reduzir esses riscos exigindo MFA e políticas de senhas fortes.

Resultado desejado: reduzir os riscos de acesso acidental a credenciais na AWS usando mecanismos de login fortes para usuários do [AWS Identity and Access Management \(IAM\)](#), o [usuário raiz da Conta da AWS](#), o [AWS IAM Identity Center](#) (sucessor do AWS Single Sign-On), e provedores de identidades de terceiros. Isso significa exigir MFA, impor políticas de senhas fortes e detectar comportamento de login anômalo.

Antipadrões comuns:

- Não impor uma política de senhas fortes para suas identidades incluindo senhas complexas e MFA.
- Compartilhar as mesmas credenciais entre usuários diferentes.
- Não utilizar controles de detecção para logins suspeitos.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientação de implementação

Há muitas formas de identidades humanas fazerem login na AWS. É prática recomendada da AWS depender de um provedor de identidades centralizado utilizando federação (federação direta ou usando AWS IAM Identity Center) ao realizar a autenticação na AWS. Nesse caso, você deve estabelecer um processo de login seguro com seu provedor de identidades ou o Microsoft Active Directory.

Ao abrir pela primeira vez uma Conta da AWS, você começa com um usuário raiz da Conta da AWS. Você só deve usar o usuário raiz da conta para configurar o acesso para seus usuários (e para [tarefas que exijam o usuário raiz](#)). É importante ativar a MFA para o usuário raiz da conta logo após abrir sua Conta da AWS e para proteger o usuário raiz usando o [Guia de práticas recomendadas da AWS](#).

Se você criar usuários no AWS IAM Identity Center, proteja o processo de login nesse serviço. Para identidades dos consumidores, é possível usar o [Amazon Cognito user pools](#) e proteger o processo de login nesse serviço ou usar os provedores de identidades compatíveis com o Amazon Cognito user pools.

Se estiver usando usuários do [AWS Identity and Access Management \(IAM\)](#), você protegerá o processo de login com o IAM.

Seja qual for o método de login, é essencial impor uma política de login forte.

Etapas da implementação

Veja a seguir as recomendações gerais de login forte. As configurações reais devem ser definidas pela política de sua empresa ou usando um padrão como [NIST 800-63](#).

- Exija MFA. É [prática recomendada IAM exigir MFA](#) para identidades humanas e workloads. A ativação da MFA oferece uma camada adicional de segurança que exige que os usuários forneçam credenciais de login e uma senha de uso único (OTP) ou uma string gerada e verificada criptograficamente por um dispositivo de hardware.
- Imponha um comprimento mínimo de senha, que é um fator essencial da força da senha.
- Imponha complexidade para tornar as senhas mais difíceis de deduzir.
- Permita que os usuários alterem suas próprias senhas.
- Crie identidades individuais em vez de credenciais compartilhadas. Com a criação de identidades individuais, é possível fornecer a cada usuário um conjunto exclusivo de credenciais de segurança. Os usuários individuais oferecem a capacidade de auditar a atividade de cada usuário.

Recomendações do IAM Identity Center:

- O IAM Identity Center oferece uma [política de senha](#) predefinida ao usar o diretório padrão que estabelece o comprimento da senha, a complexidade e requisitos de reutilização.
- [Ative a MFA](#) e defina a configuração de reconhecimento de contexto ou sempre ativo da MFA quando a origem da identidade for o diretório padrão, o AWS Managed Microsoft AD ou o AD Connector.
- Permita que os usuários [registrem seus próprios dispositivos de MFA](#).

Recomendações de diretório do Amazon Cognito user pools:

- Defina as configurações de [força da senha](#).
- [Exija MFA](#) dos usuários.
- Use as [configurações de segurança avançadas](#) do Amazon Cognito user pools para recursos como [autenticação adaptável](#) que podem bloquear logins suspeitos.

Recomendações para usuários do IAM:

- Em teoria, você está utilizando IAM Identity Center ou federação direta. No entanto, talvez você precise de usuários do IAM. Nesse caso, [defina uma política de senha](#) para usuários do IAM. A política de senhas pode ser usada para definir requisitos como extensão mínima ou a obrigatoriedade de uso de caracteres não alfabéticos.
- Crie uma política do IAM com o objetivo de [impor login de MFA](#) para que os usuários possam gerenciar suas próprias senhas e dispositivos de MFA.

Recursos

Práticas recomendadas relacionadas:

- [SEC02-BP03 Armazenar e usar segredos com segurança](#)
- [SEC02-BP04 Contar com um provedor de identidades centralizado](#)
- [SEC03-BP08 Compartilhar recursos com segurança em sua organização](#)

Documentos relacionados:

- [Política de senha do AWS IAM Identity Center \(sucessor do AWS Single Sign-On\)](#)
- [Política de senha do usuário do IAM](#)
- [Definir a senha do usuário raiz da Conta da AWS](#)
- [Política de senha do Amazon Cognito](#)
- [Credenciais da AWS](#)
- [Práticas recomendadas de segurança no IAM](#)

Vídeos relacionados:

- [Gerenciar permissões de usuário em grande escala com o AWS IAM Identity Center](#)

- [Dominar a identidade em todos os aspectos](#)

SEC02-BP02 Usar credenciais temporárias

Ao realizar qualquer tipo de autenticação, é melhor utilizar credenciais temporárias em vez de credenciais de longo prazo a fim de reduzir ou eliminar riscos, como credenciais que são divulgadas acidentalmente, compartilhadas ou roubadas.

Resultado desejado: para reduzir o risco de credenciais de longo prazo, use credenciais temporárias sempre que possível para identidades humanas e de máquina. Credenciais de longo prazo criam muitos riscos, por exemplo, é possível fazer upload delas em código para repositórios públicos do GitHub. Ao utilizar credenciais temporárias, você reduz significativamente as chances de comprometimento das credenciais.

Antipadrões comuns:

- Desenvolvedores que usam chaves de acesso de longo prazo de IAM users em vez de obter credenciais temporárias da CLI usando federação.
- Desenvolvedores que incorporam chaves de acesso de longo prazo no código e fazem upload desse código para repositórios públicos do Git.
- Desenvolvedores que incorporam chaves de acesso de longo prazo em aplicações móveis que, depois, são disponibilizadas em lojas de aplicações.
- Usuários que compartilham chaves de acesso de longo prazo com outros usuários ou funcionários que deixam a empresa e não devolvem as chaves de acesso de longo prazo.
- Utilizar chaves de acesso de longo prazo para identidades de máquina quando é possível usar credenciais temporárias.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientação de implementação

Utilize credenciais de segurança temporárias em vez de credenciais de longo prazo para todas as solicitações da AWS CLI e API. As solicitações de API e CLI para serviços da AWS devem, em quase todos os casos, ser assinadas com [chaves de acesso da AWS](#). Essas solicitações podem ser assinadas com credenciais temporárias ou de longo prazo. A única vez que você deve usar credenciais de longo prazo, também conhecidas como chaves de acesso de longo prazo, é se você estiver utilizando um [usuário do IAM](#) ou o [usuário raiz da Conta da AWS](#). Quando você usa

federação na AWS ou assume um [perfil do IAM](#) por outros métodos, são geradas credenciais temporárias. Mesmo quando você acessa o AWS Management Console utilizando credenciais de login, credenciais temporárias são geradas para você fazer chamadas para serviços da AWS. Há poucas situações nas quais você precisa de credenciais de longo prazo, e é possível realizar quase todas as tarefas usando credenciais temporárias.

Evitar o uso de credenciais de longo prazo em favor de credenciais temporárias deve andar lado a lado com uma estratégia de reduzir o uso de usuários do IAM em favor da federação e de perfis do IAM. Embora usuários do IAM tenham sido usados para identidades humanas e de máquina no passado, agora recomendamos não utilizá-los para evitar os riscos de utilizar chaves de acesso de longo prazo.

Etapas da implementação

Para identidades humanas, como funcionários, administradores, desenvolvedores, operadores e clientes:

- Você deve [contar com um provedor de identidades centralizado](#) e [exigir que usuários humanos usem federação com um provedor de identidades para acessar a AWS utilizando credenciais temporárias](#). A federação para seus usuários pode ser realizada com [federação direta para cada Conta da AWS](#) ou usando o [AWS IAM Identity Center \(sucessor do AWS IAM Identity Center\)](#) e o provedor de identidades de sua escolha. A federação oferece uma série de vantagens em comparação com a utilização de usuários do IAM além de eliminar credenciais de longo prazo. Seus usuários também podem solicitar credenciais temporárias da linha de comando para [federação direta](#) ou utilizando o [IAM Identity Center](#). Isso significa que há poucos casos de uso que exigem usuários do IAM ou credenciais de longo prazo para seus usuários.
- Ao conceder acesso a recursos em sua Conta da AWS a terceiros, como provedores de software como serviço (SaaS), você pode utilizar [perfis entre contas](#) e [políticas baseadas em recursos](#).
- Se você precisar conceder a aplicações de consumidores ou clientes acesso aos seus recursos da AWS, você pode utilizar [grupos de identidade do Amazon Cognito](#) ou [Amazon Cognito user pools](#) para fornecer credenciais temporárias. As permissões para as credenciais são configuradas por meio de perfis do IAM. Você também pode definir um perfil do IAM separado com permissões limitadas para usuários convidados que não são autenticados.

Para identidades de máquina, talvez seja necessário utilizar credenciais de longo prazo. Nesses casos, você deve [exigir que as workloads utilizem credenciais temporárias com perfis da IAM para acessar a AWS](#).

- Para [Amazon Elastic Compute Cloud](#) (Amazon EC2), é possível usar [perfis do Amazon EC2](#).
- O [AWS Lambda](#) permite configurar um [perfil de execução do Lambda para conceder permissões de serviço](#) a fim de executar ações da AWS usando credenciais temporárias. Há muitos outros modelos semelhantes para os serviços da AWS concederem credenciais temporárias utilizando perfis do IAM.
- Para serviços de IoT, é possível usar o [provedor de credenciais de AWS IoT Core](#) para solicitar credenciais temporárias.
- Para sistemas on-premises ou sistemas executados fora da AWS que precisem acessar os recursos da AWS, é possível utilizar o [IAM Roles Anywhere](#).

Há cenários em que credenciais temporárias não são uma opção e talvez seja necessário usar credenciais de longo prazo. Nessas situações, [faça auditoria e alterne as credenciais periodicamente](#) e [alterne as chaves de acesso regularmente para casos de uso que exijam credenciais de longo prazo](#). Alguns exemplos que podem exigir credenciais de longo prazo incluem plug-ins do WordPress e clientes da AWS de terceiros. Em situações em que você precisa utilizar credenciais de longo prazo ou para credenciais que não sejam chaves de acesso da AWS, como logins de banco de dados, é possível usar um serviço projetado para lidar com o gerenciamento de segredos, como [AWS Secrets Manager](#). O Secrets Manager torna simples gerenciar, alternar e armazenar com segurança segredos criptografados utilizando [serviços compatíveis](#). Para ter mais informações sobre a alternância de credenciais de longo prazo, consulte [Alternar chave de acesso](#).

Recursos

Práticas recomendadas relacionadas:

- [SEC02-BP03 Armazenar e usar segredos com segurança](#)
- [SEC02-BP04 Contar com um provedor de identidades centralizado](#)
- [SEC03-BP08 Compartilhar recursos com segurança em sua organização](#)

Documentos relacionados:

- [Credenciais de segurança temporárias](#)
- [Credenciais da AWS](#)
- [Práticas recomendadas de segurança no IAM](#)

- [Perfis do IAM](#)
- [IAM Identity Center](#)
- [Provedores de identidades e federação](#)
- [Alternar chaves de acesso](#)
- [Soluções de parceiros de segurança: acesso e controle de acesso](#)
- [Usuário raiz da Conta da AWS](#)

Vídeos relacionados:

- [Gerenciar permissões de acesso em grande escala com o AWS IAM Identity Center \(sucessor do AWS IAM Identity Center\)](#)
- [Dominar a identidade em todos os aspectos](#)

SEC02-BP03 Armazenar e usar segredos com segurança

Uma workload exige um recurso automatizado para comprovar a identidade dela em bancos de dados, recursos e serviços de terceiros. Isso é realizado com o uso de credenciais de acesso secretas, como chaves de acesso de API, senhas e tokens do OAuth. Utilizar um serviço com propósito específico para armazenar, gerenciar e alternar essas credenciais ajuda a reduzir a probabilidade de comprometimento dessas credenciais.

Resultado desejado: implementar um mecanismo para gerenciar com segurança credenciais de aplicações que concretize os seguintes objetivos:

- Identificar quais segredos são necessários para a workload.
- Reduzir o número de credenciais de longo prazo necessárias substituindo-as por credenciais de curto prazo quando possível.
- Estabelecer um armazenamento seguro e uma alternância automatizada das credenciais de longo prazo restantes.
- Auditar o acesso aos segredos existentes na workload.
- Monitorar continuamente para confirmar que nenhum segredo seja incorporado a código-fonte durante o processo de desenvolvimento.
- Reduzir a probabilidade de divulgação acidental de credenciais.

Antipadrões comuns:

- Ausência de alternância de credenciais.
- Armazenar credenciais de longo prazo em código-fonte ou arquivos de configuração.
- Armazenar credenciais em repouso não criptografadas.

Benefícios do estabelecimento desta prática recomendada:

- Os segredos são armazenados com criptografia em repouso e em trânsito.
- O acesso às credenciais é fechado por meio de uma API (pense nisso como uma máquina automática de venda de credenciais).
- O acesso a uma credencial (de leitura e gravação) é auditado e registrado.
- Separação de preocupações: a alternância de credenciais é realizada por um componente separado, que pode ser segregado do restante da arquitetura.
- Os segredos são automaticamente distribuídos sob demanda em componentes de software e a alternância ocorre em um local central.
- O acesso às credenciais pode ser controlado de forma detalhada.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientação de implementação

Antes, as credenciais usadas para realizar a autenticação em bancos de dados, APIs de terceiros, tokens e outros segredos podiam ser incorporadas em código-fonte ou em arquivos do ambiente. A AWS oferece vários mecanismos para armazenar essas credenciais com segurança, alterná-las automaticamente e auditar o uso delas.

A melhor forma de abordar o gerenciamento de segredos é seguir as orientações de remover, substituir e alternar. A credencial mais segura é a que você não precisa armazenar, gerenciar nem processar. Pode haver credenciais que não sejam mais necessárias ao funcionamento da workload que podem ser removidas com segurança.

Para credenciais que ainda são necessárias ao funcionamento adequado da workload, pode haver uma oportunidade de substituir uma credencial de longo prazo por uma credencial temporária ou de curto prazo. Por exemplo, em vez de codificar uma chave de acesso secreta da AWS, pense em substituir essa credencial de longo prazo por uma temporária utilizando perfis do IAM.

Alguns segredos duradouros podem não ser removidos ou substituídos. Esses segredos podem ser armazenados em um serviço, como o [AWS Secrets Manager](#), no qual eles podem ser armazenados centralmente, gerenciados e alternados regularmente.

Uma auditoria do código-fonte da workload e os arquivos de configuração podem revelar muitos tipos de credencial. A seguinte tabela resume as estratégias para lidar com tipos comuns de credenciais:

Credential type	Description	Suggested strategy
IAM access keys	AWS IAM access and secret keys used to assume IAM roles inside of a workload	Replace: Use Perfis do IAM assigned to the compute instances (such as Amazon EC2 or AWS Lambda) instead. For interoperability with third parties that require access to resources in your Conta da AWS, ask if they support Acesso entre contas da AWS . For mobile apps, consider using temporary credentials through Grupos de identidad es (identidades federadas) do Amazon Cognito . For workloads running outside of AWS, consider IAM Roles Anywhere or AWS Systems Manager Hybrid Activations .
SSH keys	Secure Shell private keys used to log into Linux EC2 instances, manually or as part of an automated process	Replace: Use AWS Systems Manager or EC2 Instance Connect to provide programmatic and human access to EC2 instances using IAM roles.
Application and database credentials	Passwords – plain text string	Rotate: Store credentials in AWS Secrets Manager and

Credential type	Description	Suggested strategy
		establish automated rotation if possible.
Amazon RDS and Aurora Admin Database credentials	Passwords – plain text string	Replace: Use the Integração do Secrets Manager ao Amazon RDS or Amazon Aurora . In addition, some RDS database types can use IAM roles instead of passwords for some use cases (for more detail, see Autenticação de banco de dados do IAM).
OAuth tokens	Secret tokens – plain text string	Rotate: Store tokens in AWS Secrets Manager and configure automated rotation.
API tokens and keys	Secret tokens – plain text string	Rotate: Store in AWS Secrets Manager and establish automated rotation if possible.

Um antipadrão comum é incorporar chaves de acesso do IAM ao código-fonte, a arquivos de configuração ou aplicativos móveis. Quando uma chave de acesso do IAM é necessária para comunicação com um serviço da AWS, utilize [credenciais de segurança temporárias \(de curto prazo\)](#). Essas credenciais de curto prazo podem ser fornecidas por meio de [perfis do IAM para instâncias do EC2](#), [perfis de execução](#) para funções do Lambda, [perfis do Cognito IAM](#) para acesso de usuários móveis e [políticas do IoT Core](#) para dispositivos IoT. Ao fazer interface com terceiros, prefira [delegar o acesso a um perfil do IAM](#) com o acesso necessário aos recursos de sua conta em vez de configurar um usuário do IAM e enviar a terceiros a chave de acesso secreta desse usuário.

Há muitos casos em que a workload exige o armazenamento de segredos necessários para interoperar com outros serviços e recursos. O [AWS Secrets Manager](#) foi concebido especificamente para gerenciar com segurança essas credenciais, bem como o armazenamento, o uso e a alternância de tokens de API, senhas e outras credenciais.

O AWS Secrets Manager oferece cinco recursos principais para proteger o armazenamento e o processamento de credenciais sigilosas: [criptografia em repouso](#), [criptografia em trânsito](#), [auditoria abrangente](#), [controle de acesso detalhado](#) e [alternância de credenciais extensíveis](#). Outros serviços de gerenciamento de segredos de parceiros da AWS ou soluções desenvolvidas localmente que oferecem recursos e garantias semelhantes também são aceitáveis.

Etapas da implementação

1. Identifique caminhos de código que contenham credenciais codificadas usando ferramentas automatizadas, como o [Amazon CodeGuru](#).
 - Utilize o Amazon CodeGuru para verificar seus repositórios de código. Depois de concluir a revisão, filtre Type=Secrets no CodeGuru para encontrar linhas de código problemáticas.
2. Identifique credenciais que possam ser removidas ou substituídas.
 - a. Identifique credenciais não mais necessárias e marque-as para remoção.
 - b. Para chaves secretas da AWS incorporadas ao código-fonte, substitua-as por perfis do IAM associados aos recursos necessários. Se parte de sua workload estiver fora do AWS, mas exigir credenciais do IAM para acessar recursos da AWS, considere o [IAM Roles Anywhere](#) ou o [AWS Systems Manager Hybrid Activations](#).
3. Para outros terceiros, segredos duradouros que exijam o uso da estratégia de alternância, integre o Secrets Manager ao seu código para recuperar segredos de terceiros no tempo de execução.
 - a. O console do CodeGuru pode [criar um segredo automaticamente no Secrets Manager](#) utilizando as credenciais descobertas.
 - b. Integre a recuperação de segredos do Secrets Manager ao código de sua aplicação.
 - Funções do Lambda Sem Servidor podem usar uma [extensão do Lambda](#) independente de linguagem.
 - Para instâncias ou contêineres do EC2, a AWS oferece [código do lado do cliente de exemplo para recuperar segredos do Secrets Manager](#) em várias linguagens de programação conhecidas.
4. Revise periodicamente sua base de código e verifique novamente para confirmar se não há novos segredos adicionados ao código.
 - Considere usar uma ferramenta como o [git-secrets](#) para impedir a confirmação de novos segredos em seu repositório de código-fonte.
5. [Monitore a atividade do Secrets Manager](#) quanto a indicações de uso inesperado, acesso inadequado a segredos ou tentativas de excluir segredos.

6. Reduza a exposição humana às credenciais. Restrinja o acesso a credenciais de leitura, gravação e modificação a um perfil do IAM dedicado a esse fim, e apenas forneça acesso para assumir o perfil a um pequeno subconjunto de usuários operacionais.

Recursos

Práticas recomendadas relacionadas:

- [SEC02-BP02 Usar credenciais temporárias](#)
- [SEC02-BP05 Fazer a auditoria e a alternância periódica das credenciais](#)

Documentos relacionados:

- [Conceitos básicos do AWS Secrets Manager](#)
- [Provedores de identidades e federação](#)
- [Amazon CodeGuru apresenta o Secrets Detector](#)
- [Como o AWS Secrets Manager usa o AWS Key Management Service](#)
- [Criptografia e descriptografia de segredos no Secrets Manager](#)
- [Entradas do blog do Secrets Manager](#)
- [Amazon RDS anuncia integração com o AWS Secrets Manager](#)

Vídeos relacionados:

- [Práticas recomendadas para gerenciar, recuperar e alternar segredos em grande escala](#)
- [Encontre segredos codificados com o Amazon CodeGuru Secrets Detector](#)
- [Proteger segredos para workloads híbridas usando o AWS Secrets Manager](#)

Workshops relacionados:

- [Armazenar, recuperar e gerenciar credenciais sigilosas no AWS Secrets Manager](#)
- [Ativações híbridas do AWS Systems Manager](#)

SEC02-BP04 Contar com um provedor de identidades centralizado

Para identidades da força de trabalho (funcionários e prestadores de serviços), confie em um provedor de identidade que permita gerenciar identidades em um local centralizado. Isso facilita o gerenciamento do acesso em várias aplicações e sistemas, pois você está criando, atribuindo, gerenciando, revogando e auditando o acesso de um único local.

Resultado desejado: Você tem um provedor de identidade centralizado no qual gerencia centralmente os usuários da força de trabalho, as políticas de autenticação (como a exigência de autenticação multifator (MFA)) e a autorização para sistemas e aplicações (como atribuir acesso com base na associação ou nos atributos do grupo de um usuário). Os usuários da sua força de trabalho fazem login no provedor de identidade central e se federam (autenticação única) a aplicações internas e externas, eliminando a necessidade de os usuários se lembrarem de várias credenciais. Seu provedor de identidade é integrado aos seus sistemas de recursos humanos (RH) para que as mudanças de pessoal sejam automaticamente sincronizadas com seu provedor de identidade. Por exemplo, se alguém deixar sua organização, você poderá revogar automaticamente o acesso a aplicações e sistemas federados (inclusive a AWS). Você habilitou o registro em log detalhado de auditoria em seu provedor de identidade e está monitorando esses logs em busca de comportamentos incomuns do usuário.

Antipadrões comuns:

- Você não usa federação e autenticação única. Os usuários da sua força de trabalho criam contas de usuário e credenciais separadas em várias aplicações e sistemas.
- Você não automatizou o ciclo de vida das identidades dos usuários da força de trabalho, por exemplo, integrando seu provedor de identidade aos seus sistemas de RH. Quando um usuário deixa sua organização ou muda de função, você segue um processo manual para excluir ou atualizar seus registros em várias aplicações e sistemas.

Benefícios de estabelecer esta prática recomendada: Ao usar um provedor de identidade centralizado, você tem um único local para gerenciar as identidades e políticas dos usuários da força de trabalho, a capacidade de atribuir acesso às aplicações a usuários e grupos e a capacidade de monitorar a atividade de login do usuário. Ao se integrar aos seus sistemas de recursos humanos (RH), quando um usuário muda de função, essas alterações são sincronizadas com o provedor de identidade e atualizam automaticamente as aplicações e permissões atribuídas. Quando um usuário sai da sua organização, sua identidade é automaticamente desativada no provedor de identidade, revogando seu acesso a aplicações e sistemas federados.

Nível de risco exposto se esta prática recomendada não for estabelecida: alto

Orientação para implementação

Orientação para usuários da força de trabalho que acessam a AWS

Usuários da força de trabalho em sua organização, como funcionários e prestadores de serviços, podem precisar acessar a AWS usando o AWS Management Console ou a AWS Command Line Interface (AWS CLI) para realizar suas funções de trabalho. Você pode conceder acesso à AWS aos usuários da sua força de trabalho federando a partir de seu provedor de identidade centralizado para a AWS em dois níveis: federação direta para cada Conta da AWS ou federação para várias contas na [organização da AWS](#).

- Para federar os usuários da sua força de trabalho diretamente com cada Conta da AWS, você pode usar um provedor de identidade centralizado para federar o [AWS Identity and Access Management](#) nessa conta. A flexibilidade do IAM permite que você habilite um [SAML 2.0](#) ou um [provedor de identidade Open ID Connect \(OIDC\)](#) para cada Conta da AWS e use atributos de usuário federados para controle de acesso. Os usuários da sua força de trabalho usarão o navegador da web para fazer login no provedor de identidade fornecendo suas respectivas credenciais (como senhas e códigos de token MFA). O provedor de identidade emite uma declaração SAML para o navegador, que é enviada ao URL de login do AWS Management Console para permitir que o usuário faça autenticação única no [AWS Management Console assumindo uma função do IAM](#). Seus usuários também podem obter credenciais temporárias de API da AWS para uso na [AWS CLI](#) ou [em AWS SDKs](#) pelo [AWS STS](#) assumindo [a função do IAM usando uma declaração SAML](#) do provedor de identidade.
- Para federar seus usuários da força de trabalho com várias contas em sua organização da AWS, você pode usar o [AWS IAM Identity Center](#) para gerenciar centralmente o acesso dos usuários de sua força de trabalho a Contas da AWS e aplicações. Você ativa o Centro de Identidade para sua organização e configura sua fonte de identidade. O IAM Identity Center fornece um diretório de origem de identidade padrão que você pode usar para gerenciar seus usuários e grupos. Como alternativa, você pode escolher uma fonte de identidade externa [conectando-se ao seu provedor de identidade externo](#) usando SAML 2.0 e [provisionando automaticamente](#) usuários e grupos usando o SCIM ou [conectando-se ao diretório do Microsoft AD](#) com o uso do [AWS Directory Service](#). Depois que uma fonte de identidade é configurada, você pode atribuir acesso a usuários e grupos a Contas da AWS definindo políticas de privilégios mínimos em seus [conjuntos de permissões](#). Os usuários da sua força de trabalho podem se autenticar por meio de seu provedor de identidade central para entrar no [portal de acesso da AWS](#) e autenticação única em Contas da AWS e aplicações em nuvem atribuídas a eles. Seus usuários podem configurar a [AWS CLI v2](#)

para se autenticar com o Centro de Identidade e obter credenciais para executar comandos da AWS CLI. O Centro de Identidade também permite acesso com autenticação única a aplicações da AWS, como o [Amazon SageMaker Studio](#) e [portais do AWS IoT Sitewise Monitor](#).

Depois de seguir as orientações anteriores, os usuários da sua força de trabalho não precisarão mais usar IAM users e grupos para operações normais ao gerenciar workloads na AWS. Em vez disso, seus usuários e grupos são gerenciados fora da AWS e os usuários podem acessar os recursos da AWS como uma identidade federada. As identidades federadas usam os grupos definidos pelo seu provedor de identidade centralizado. Você deve identificar e remover grupos do IAM, IAM users e credenciais de usuário de longa duração (senhas e chaves de acesso) que não são mais necessárias nas suas Contas da AWS. Você pode [encontrar credenciais não utilizadas](#) com o uso do [relatórios de credenciais do IAM](#), [excluindo IAM users correspondentes](#) e [excluindo grupos do IAM](#). Você pode aplicar uma [política de controle de serviços \(SCP\)](#) na sua organização, o que ajudará a impedir a criação de novos grupos e IAM users, forçando que esse acesso ocorra por meio de identidades federadas da AWS.

Orientação para usuários de suas aplicações

Você pode gerenciar as identidades dos usuários de suas aplicações, como um aplicativo móvel, usando o [Amazon Cognito](#) como seu provedor de identidade centralizado. O Amazon Cognito permite autenticação, autorização e gerenciamento de usuários de seus aplicativos móveis e da web. O Amazon Cognito fornece um repositório de identidades que pode ser expandido para milhões de usuários, oferece suporte à federação de identidades sociais e corporativas e oferece recursos avançados de segurança para ajudar a proteger seus usuários e sua empresa. Você pode integrar seu aplicativo web ou móvel personalizado ao Amazon Cognito para adicionar autenticação de usuário e controle de acesso aos seus aplicativos em minutos. Desenvolvido com base em padrões de identidade abertos, como SAML e Open ID Connect (OIDC), o Amazon Cognito oferece suporte a vários regulamentos de conformidade e se integra aos recursos de desenvolvimento de front-end e back-end.

Etapas da implementação

Etapas para usuários da força de trabalho acessarem a AWS

- Federe os usuários da sua força de trabalho à AWS usando um provedor de identidade centralizado de acordo com uma das seguintes abordagens:
 - Use o IAM Identity Center para habilitar a autenticação única para várias Contas da AWS em sua organização da AWS, federando com seu provedor de identidade.

- Use o IAM para conectar seu provedor de identidade diretamente a cada Conta da AWS, permitindo acesso federado refinado.
- Identifique e remova IAM users e grupos que são substituídos por identidades federadas.

Etapas para usuários de suas aplicações

- Use o Amazon Cognito como um provedor de identidade centralizado para suas aplicações.
- Integre suas aplicações personalizadas com o Amazon Cognito usando o OpenID Connect e o OAuth. Você pode desenvolver suas aplicações personalizadas usando as bibliotecas do Amplify que fornecem interfaces simples para integração com uma variedade de serviços da AWS, como o Amazon Cognito para autenticação.

Recursos

Práticas recomendadas relacionadas ao Well-Architected:

- [SEC02-BP06 Utilizar grupos de usuários e atributos](#)
- [SEC03-BP02 Conceder acesso com privilégio mínimo](#)
- [SEC03-BP06 Gerenciar o acesso com base no ciclo de vida](#)

Documentos relacionados:

- [Federação de identidades na AWS](#)
- [Práticas recomendadas de segurança no IAM](#)
- [Práticas recomendadas do AWS Identity and Access Management](#)
- [Introdução à administração delegada do IAM Identity Center](#)
- [Como usar políticas gerenciadas pelo cliente no IAM Identity Center para casos de uso avançados](#)
- [AWS CLI v2: provedor de credenciais do IAM Identity Center](#)

Vídeos relacionados:

- [AWS re:Inforce 2022 - AWS Identity and Access Management \(IAM\) deep dive \(AWS re:Inforce 2022: aprofundamento no AWS Identity and Access Management \(IAM\)\)](#)
- [AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center \(AWS re:Invent 2022: simplifique o acesso existente de sua força de trabalho com o IAM Identity Center\)](#)

- [AWS re:Invent 2018: Mastering Identity at Every Layer of the Cake \(AWS re:Invent 2018: dominar a identidade em todos os aspectos\)](#)

Exemplos relacionados:

- [Workshop: Using AWS IAM Identity Center to achieve strong identity management \(Uso do AWS IAM Identity Center para conseguir um forte gerenciamento de identidade\)](#)
- [Workshop: Serverless identity \(Identidade sem servidor\)](#)

Ferramentas relacionadas:

- [Parceiros de competência em segurança da AWS: gerenciamento de identidade e acesso](#)
- [saml2aws](#)

SEC02-BP05 Fazer a auditoria e a alternância periódica das credenciais

Audite e alterne as credenciais periodicamente para limitar o período durante o qual as credenciais podem ser usadas para acessar seus recursos. Credenciais de longo prazo criam muitos riscos, e estes podem ser reduzidos alternando credenciais de longo prazo regularmente.

Resultado desejado: implementar a alternância de credenciais para ajudar a reduzir os riscos associados ao uso de credenciais de longo prazo. Auditar e corrigir regularmente a não conformidade com políticas de alternância de credenciais.

Antipadrões comuns:

- Não auditar o uso de credenciais.
- Utilizar credenciais de longo prazo desnecessariamente.
- Utilizar credenciais de longo prazo e não alterná-las regularmente.

Nível de risco exposto se esta prática recomendada não é estabelecida: médio

Orientação de implementação

Quando você não puder contar com credenciais temporárias e exigir credenciais de longo prazo, faça uma auditoria das credenciais para garantir que os controles definidos, por exemplo,

autenticação multifator (MFA), sejam aplicados, alternados regularmente e que tenham o nível de acesso apropriado.

A validação periódica, preferencialmente por meio de uma ferramenta automatizada, é necessária para verificar se os controles corretos são aplicados. Para identidades humanas, você deve exigir que os usuários alterem suas senhas periodicamente e substituam chaves de acesso por credenciais temporárias. Ao migrar de usuários do AWS Identity and Access Management (IAM) para identidades centralizadas, é possível [gerar um relatório de credenciais](#) para fazer auditoria de seus usuários.

Também recomendamos implementar e monitorar a MFA no provedor de identidades. É possível configurar o [Regras do AWS Config](#) ou usar [Padrões de segurança AWS Security Hub](#) para monitorar se os usuários têm a MFA ativada. Considere utilizar o IAM Roles Anywhere para fornecer credenciais temporárias para identidades de máquina. Em situações em que o uso de perfis do IAM e credenciais temporárias não é possível, é necessário realizar auditoria frequente e alternar as chaves de acesso.

Etapas da implementação

- Fazer auditoria nas credenciais regularmente: a auditoria das identidades configuradas em seu provedor de identidades e no IAM ajuda a garantir que somente identidades autorizadas tenham acesso à sua workload. Essas identidades podem incluir, entre outros, usuários do IAM, do AWS IAM Identity Center, do Active Directory ou usuários em um provedor de identidades upstream diferente. Por exemplo, remova as pessoas que saem da organização e as funções entre contas que não são mais necessárias. Estabeleça um processo para auditar periodicamente as permissões para os serviços acessados por uma entidade do IAM. Isso ajuda a identificar as políticas que você precisa modificar a fim de remover todas as permissões não utilizadas. Use relatórios de credenciais e o [AWS Identity and Access Management Access Analyzer](#) para auditar credenciais e permissões do IAM. É possível utilizar o [Amazon CloudWatch para configurar alarmes para chamadas de API específicas](#) chamadas em seu ambiente da AWS. [O Amazon GuardDuty também pode alertar você sobre atividade inesperada](#), que pode indicar acesso excessivamente permissivo ou acesso acidental às credenciais do IAM.
- Alternar credenciais regularmente: quando você não pode utilizar credenciais temporárias, alterne as chaves de acesso do IAM de longo prazo regularmente (no máximo, a cada 90 dias). Se uma chave de acesso for divulgada acidentalmente sem seu conhecimento, isso limitará o período de uso das credenciais para acessar seus recursos. Para ter informações sobre a alternância de chaves de acesso para usuários do IAM, consulte [Alternar chaves de acesso](#).

- Revisar as permissões do IAM: para melhorar a segurança de sua Conta da AWS, revise e monitore regularmente cada uma das políticas do IAM. Verifique se as políticas seguem o princípio de privilégio mínimo.
- Considerar automatizar a criação e as atualizações dos recursos do IAM: o IAM Identity Center automatiza muitas tarefas do IAM, como o gerenciamento de perfis e políticas. Como alternativa, o AWS CloudFormation pode ser usado para automatizar a implantação de recursos do IAM, como perfis e políticas, para reduzir a chance de erros humanos, pois os modelos podem ser verificados e ter controle de versão.
- Utilizar o IAM Roles Anywhere para substituir os usuários do IAM para identidades de máquina: o IAM Roles Anywhere possibilita usar perfis em áreas onde não seria possível tradicionalmente, como em servidores on-premises. O IAM Roles Anywhere utiliza um certificado X.509 confiável para realizar a autenticação na AWS e receber credenciais temporárias. O uso do IAM Roles Anywhere evita a necessidade de alternar essas credenciais, pois credenciais de longo prazo não são mais armazenadas em seu ambiente on-premises. Você precisará monitorar e alternar o certificado X.509 ao aproximar-se da validade.

Recursos

Práticas recomendadas relacionadas:

- [SEC02-BP02 Usar credenciais temporárias](#)
- [SEC02-BP03 Armazenar e usar segredos com segurança](#)

Documentos relacionados:

- [Conceitos básicos do AWS Secrets Manager](#)
- [Práticas recomendadas do IAM](#)
- [Provedores de identidades e federação](#)
- [Soluções de parceiros de segurança: acesso e controle de acesso](#)
- [Credenciais de segurança temporárias](#)
- [Obter relatórios de credenciais da sua Conta da AWS](#)

Vídeos relacionados:

- [Práticas recomendadas para gerenciar, recuperar e alternar segredos em grande escala](#)

- [Gerenciar permissões de usuário em grande escala com o AWS IAM Identity Center](#)
- [Dominar a identidade em todos os aspectos](#)

Exemplos relacionados:

- [Well-Architected Lab: Limpeza automatizada de usuários do IAM](#)
- [Well-Architected Lab: Implantação automatizada de grupos e perfis do IAM](#)

SEC02-BP06 Utilizar grupos de usuários e atributos

A definição de permissões de acordo com grupos de usuários e atributos ajuda a reduzir o número e a complexidade das políticas, simplificando o cumprimento do princípio do privilégio mínimo. Você pode usar grupos de usuários para gerenciar permissões a várias pessoas em um só lugar com base na função que elas desempenham em sua organização. Os atributos, como departamento ou localização, podem ampliar o escopo de permissão quando as pessoas realizam uma função que, embora semelhante, envolve diferentes subconjuntos de recursos.

Resultado desejado: é possível aplicar alterações nas permissões com base na função a todos os usuários que desempenham essa função. A associação de grupo e os atributos governam as permissões de usuário, reduzindo a necessidade de gerenciar permissões para cada usuário. Os grupos e atributos que você define em seu provedor de identidades (IdP) são propagados automaticamente para seus ambientes da AWS.

Antipadrões comuns:

- Gerenciar permissões para usuários individuais e duplicá-las entre vários usuários.
- Definir grupos de uma forma muito genérica, concedendo permissões excessivamente amplas.
- Definir grupos em um nível muito detalhado, criando duplicação e confusão em termos de associação.
- Usar grupos com permissões duplicadas em subconjuntos de recursos quando, em vez disso, é possível usar atributos.
- Não gerenciar grupos, atributos e associações por meio de um provedor de identidades padronizado integrado aos seus ambientes da AWS.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

As permissões da AWS são definidas em documentos chamados políticas, as quais estão associadas a uma entidade principal, como usuário, grupo, função ou recurso. Para seu quadro de funcionários, isso permite que você defina grupos com base na função que os usuários realizam dentro da organização, e não com base nos recursos que estão sendo acessados. Por exemplo, um grupo `WebAppDeveloper` pode ter uma política anexada para configurar um serviço (como o Amazon CloudFront) em uma conta de desenvolvimento. Um grupo `AutomationDeveloper` pode ter algumas permissões do CloudFront em comum com o grupo `WebAppDeveloper`. Essas permissões podem ser capturadas em uma política separada e associadas aos dois grupos, em vez de atribuir usuários de ambas as funções a um grupo `CloudFrontAccess`.

Além dos grupos, você pode usar atributos para ampliar o acesso ao escopo. Por exemplo, você pode ter um atributo `Project` para usuários do grupo `WebAppDeveloper` para definir o acesso a recursos específicos do projeto. O uso dessa técnica elimina a necessidade de ter grupos diferentes para desenvolvedores de aplicações que estão trabalhando em diferentes projetos se, em outras circunstâncias, as permissões deles forem as mesmas. A forma como você se refere aos atributos nas políticas de permissão baseia-se na respectiva origem, sejam eles definidos como parte do seu protocolo de federação (por ex., SAML, OIDC ou SCIM) ou como declarações SAML personalizadas, ou definidos dentro do IAM Identity Center.

Etapas da implementação

1. Estabeleça onde você definirá grupos e atributos.
 - a. Seguindo as orientações em [SEC02-BP04 Contar com um provedor de identidades centralizado](#), você pode determinar se precisa definir grupos e atributos no provedor de identidades ou no IAM Identity Center ou usando grupos do IAM user em uma conta específica.
2. Defina grupos.
 - a. Determine seus grupos com base na função e no escopo de acesso necessário.
 - b. Se estiver definindo no IAM Identity Center, crie grupos e associe o nível de acesso desejado usando conjuntos de permissões.
 - c. Se estiver definindo em um provedor de identidades externo, determine se o provedor atende ao protocolo SCIM e considere habilitar o provisionamento automático no IAM Identity Center. Esse recurso sincroniza a criação, associação e exclusão de grupos entre seu provedor e o IAM Identity Center.
3. Defina os atributos.

- a. Se estiver usando um provedor de identidades externo, os protocolos SCIM e SAML 2.0 fornecem determinados atributos por padrão. Atributos adicionais podem ser definidos e passados por meio de declarações SAML usando o nome de atributo `https://aws.amazon.com/SAML/Attributes/PrincipalTag`.
 - b. Se estiver definindo no IAM Identity Center, habilite o recurso de controle de acesso por atributo (ABAC) e defina os atributos conforme desejado.
4. Defina o escopo das permissões com base em grupos e atributos.
- a. Considere incluir condições em suas políticas de permissão que comparem os atributos da entidade principal com os atributos dos recursos que estão sendo acessados. Por exemplo, você pode definir uma condição para permitir acesso a um recurso somente se o valor de uma chave de condição `PrincipalTag` corresponder ao valor de uma chave `ResourceTag` com o mesmo nome.

Recursos

Práticas recomendadas relacionadas:

- [SEC02-BP04 Contar com um provedor de identidades centralizado](#)
- [SEC03-BP02 Conceder acesso com privilégio mínimo](#)
- [COST02-BP04 Implementar grupos e perfis](#)

Documentos relacionados:

- [IAM Best Practices](#)
- [Manage Identities in IAM Identity Center](#)
- [What Is ABAC for AWS?](#)
- [ABAC In IAM Identity Center](#)

Vídeos relacionados:

- [Managing user permissions at scale with AWS IAM Identity Center](#)
- [Mastering identity at every layer of the cake](#)

Gerenciamento de permissões

Gerencie permissões para controlar o acesso a identidades de humanos e máquinas que precisam de acesso à AWS e à suas workloads. As permissões controlam quem pode acessar o quê e em quais condições. Defina permissões para identidades humanas e de máquina específicas para conceder acesso a ações de serviço específicas em recursos específicos. Além disso, especifique condições que devem ser verdadeiras para que o acesso seja concedido. Por exemplo, você pode permitir que os desenvolvedores criem novas funções do Lambda, mas apenas em uma Região específica. Ao gerenciar seus ambientes da AWS em escala, siga as práticas recomendadas a seguir para garantir que as identidades tenham apenas o acesso de que precisam e nada mais.

Há várias maneiras de conceder acesso a diferentes tipos de recursos. Uma maneira é usar diferentes tipos de política.

[Políticas baseadas em identidade](#) no IAM são gerenciadas ou em linha, e se agregam a identidades do IAM, incluindo usuários, grupos e funções. Essas políticas permitem especificar o que essa identidade pode fazer (permissões). As políticas baseadas em identidade podem ser subdivididas em outras categorias.

Políticas gerenciadas: políticas individuais baseadas em identidade que você pode anexar a vários usuários, grupos e funções em sua conta da AWS. Há dois tipos de políticas gerenciadas:

- Políticas gerenciadas pela AWS: políticas gerenciadas que são criadas e gerenciadas pela AWS.
- Políticas gerenciadas pelo cliente: políticas gerenciadas que você cria e gerencia em sua conta da AWS. As políticas gerenciadas pelo cliente fornecem controle mais preciso sobre suas políticas do que as políticas gerenciadas pela AWS.

As políticas gerenciadas são o método preferencial para aplicar permissões. No entanto, também é possível usar políticas em linha adicionadas diretamente a um único usuário, grupo ou função. As políticas em linha mantêm uma relação um para um estrita entre uma política e uma identidade. As políticas em linha são excluídas quando você exclui a identidade.

Na maioria dos casos, é necessário criar as próprias políticas gerenciadas pelo cliente seguindo o princípio de [privilégio mínimo](#).

[Políticas baseadas em recursos](#) são anexadas ao recurso. Por exemplo, uma política de bucket do S3 é uma política baseada em recursos. Essas políticas concedem permissão a uma entidade principal que pode estar na mesma conta que o recurso ou em outra conta. Para obter uma lista de

serviços que oferecem suporte a políticas baseadas em recursos, consulte [AWS services that work with IAM](#) (Serviços da AWS que funcionam com IAM).

[Limites de permissões](#) usam uma política gerenciada para definir as permissões máximas que um administrador pode definir. Isso permite que você delegue a capacidade de criar e gerenciar permissões para desenvolvedores, como a criação de um perfil do IAM, mas limita as permissões que eles podem conceder para que não possam escalar as próprias permissões usando o que eles criaram.

[Controle de acesso baseado em atributos \(ABAC\)](#) permite que você conceda permissões com base em atributos. Na AWS, elas são chamadas de tags. As tags podem ser anexadas a entidades principais (usuários ou funções) do IAM e a recursos da AWS. Usando políticas do IAM, os administradores podem criar uma política reutilizável que aplique permissões com base nos atributos da entidade principal do IAM. Por exemplo, como administrador, você pode usar uma única política do IAM que concede aos desenvolvedores em sua organização acesso a recursos da AWS que correspondam às tags de projeto dos desenvolvedores. À medida que a equipe de desenvolvedores adiciona recursos aos projetos, as permissões são aplicadas automaticamente com base em atributos. Como resultado, nenhuma atualização de política é necessária para cada novo recurso.

[Políticas de controle de serviço \(SCP\)](#) definem as permissões máximas para membros da conta de uma organização ou unidade organizacional (UO). As SCPs limitam as permissões que as políticas baseadas em identidade ou as políticas baseadas em recursos concedem a entidades (usuários ou funções) dentro da conta, mas não concedem permissões.

[Políticas de sessão](#) assumem uma função ou um usuário federado. Passe as políticas de sessão ao usar as políticas de sessão da AWS CLI ou AWS API para limitar as permissões que as políticas baseadas em identidade do usuário ou da função concedem à sessão. Essas políticas limitam as permissões para uma sessão criada, mas não concedem. Para obter mais informações, consulte [Políticas de sessão](#).

Práticas recomendadas

- [SEC03-BP01 Definir requisitos de acesso](#)
- [SEC03-BP02 Conceder acesso com privilégio mínimo](#)
- [SEC03-BP03 Estabelecer processo de acesso de emergência](#)
- [SEC03-BP04 Reduzir as permissões continuamente](#)
- [SEC03-BP05 Definir barreiras de proteção de permissões para sua organização](#)
- [SEC03-BP06 Gerenciar o acesso com base no ciclo de vida](#)

- [SEC03-BP07 Analisar o acesso público e entre contas](#)
- [SEC03-BP08 Compartilhar recursos com segurança em sua organização](#)
- [SEC03-BP09 Compartilhar recursos com segurança com terceiros](#)

SEC03-BP01 Definir requisitos de acesso

Cada componente ou recurso de sua workload precisa ser acessado por administradores, usuários finais ou outros componentes. É necessário ter uma definição clara de quem ou do que deve ter acesso a cada componente, escolher o tipo de identidade apropriado e o método de autenticação e autorização.

Antipadrões comuns:

- Codificação rígida ou armazenamento de segredos em sua aplicação.
- Conceder permissões personalizadas a cada usuário.
- Uso de credenciais de longa duração.

Nível de risco exposto se essa prática recomendada não for estabelecida: alto

Orientação para implementação

Cada componente ou recurso de sua workload precisa ser acessado por administradores, usuários finais ou outros componentes. É necessário ter uma definição clara de quem ou do que deve ter acesso a cada componente, escolher o tipo de identidade apropriado e o método de autenticação e autorização.

O acesso regular a Contas da AWS na organização deve ser fornecido usando [acesso federado](#) ou um provedor de identidade centralizado. Você também deve centralizar o gerenciamento de identidades e garantir que haja uma prática estabelecida para integrar o acesso à AWS ao ciclo de vida de acesso dos funcionários. Por exemplo, quando um funcionário muda para um cargo com um nível de acesso diferente, sua associação ao grupo também deve mudar para refletir os novos requisitos de acesso.

Ao definir os requisitos de acesso para identidades não humanas, determine quais aplicações e componentes precisam de acesso e como as permissões são concedidas. O uso de perfis do IAM criados com o modelo de acesso de privilégio mínimo é uma abordagem recomendada. [As políticas gerenciadas pela AWS](#) fornecem políticas predefinidas do IAM que abordam a maioria dos casos de uso comuns.

Os serviços da AWS, como o [AWS Secrets Manager](#) e o [AWS Systems Manager Parameter Store](#), podem ajudar a desacoplar segredos da aplicação ou workload com segurança em casos em que não é possível usar perfis do IAM. No Secrets Manager, você pode estabelecer uma alternância automática de suas credenciais. É possível usar o Systems Manager para referenciar parâmetros em seus scripts, comandos, documentos do SSM, configurações e fluxos de trabalho de automação, usando o nome exclusivo que você especificou ao criar o parâmetro.

Você pode usar o AWS Identity and Access Management Roles Anywhere para obter [credenciais de segurança temporárias no IAM](#) para workloads executadas fora da AWS. As workloads podem usar as mesmas [políticas do IAM](#) e [perfis do IAM](#) que você usa com as aplicações da AWS para acessar os recursos da AWS.

Quando possível, prefira credenciais temporárias de curta duração em vez de credenciais estáticas de longa duração. Para cenários em que você precisa de usuários da IAM com acesso programático e credenciais de longa duração, use [as últimas informações usadas da chave de acesso](#) para alternar e remover chaves de acesso.

Recursos

Documentos relacionados:

- [Controle de acesso por atributo \(ABAC\)](#)
- [AWS IAM Identity Center](#)
- [IAM Roles Anywhere](#)
- [AWS Managed policies for IAM Identity Center \(Políticas gerenciadas pela AWS para o IAM Identity Center\)](#)
- [AWS IAM policy conditions \(Condições de políticas do AWS IAM\)](#)
- [IAM use cases \(Casos de uso do IAM\)](#)
- [Remova credenciais desnecessárias](#)
- [Trabalhando com políticas](#)
- [How to control access to AWS resources based on Conta da AWS, OU, or organization \(Como controlar o acesso aos recursos da AWS baseados em Conta da AWS, UO ou organização\)](#)
- [Identify, arrange, and manage secrets easily using enhanced search in AWS Secrets Manager \(Identificar, organizar e gerenciar segredos facilmente usando a pesquisa avançada no AWS Secrets Manager\)](#)

Vídeos relacionados:

- [Become an IAM Policy Master in 60 Minutes or Less \(Torne-se um mestre em políticas do IAM em 60 minutos ou menos\)](#)
- [Separation of Duties, Least Privilege, Delegation, and CI/CD \(Separação de tarefas, privilégio mínimo, delegação e CI/CD\)](#)
- [Streamlining identity and access management for innovation \(Simplificação do gerenciamento de identidade e acesso para inovação\)](#)

SEC03-BP02 Conceder acesso com privilégio mínimo

É prática recomendada conceder somente o acesso de que as identidades precisam para realizar ações em recursos específicos e sob condições específicas. Use grupos e atributos de identidade para definir permissões dinamicamente em escala, em vez de definir permissões para usuários individuais. Por exemplo, você pode permitir o acesso de um grupo de desenvolvedores para gerenciar apenas recursos de seu próprio projeto. Dessa forma, se um desenvolvedor sair do projeto, o acesso dele é automaticamente revogado sem alterar as políticas de acesso adjacentes.

Resultado desejado: os usuário somente têm permissões necessárias para fazerem seus respectivos trabalhos. Os usuários devem ter acesso apenas a ambientes de produção para realizar uma tarefa específica dentro de um período limitado e o acesso deve ser revogado quando a tarefa for concluída. As permissões devem ser revogadas quando não forem mais necessárias, incluindo quando um usuário for para um projeto diferente ou mudar de cargo. Privilégios de administrador devem ser concedidos apenas a um grupo pequeno de administradores confiáveis. As permissões devem ser revistas regularmente para evitar desvios de permissão. Contas de máquina ou sistema devem ter apenas o mínimo de permissões necessárias para concluir as tarefas.

Antipadrões comuns:

- Usar como padrão a concessão de permissões de administrador aos usuários.
- Usar o usuário raiz para atividades diárias.
- Criar políticas permissivas demais, mas sem privilégios completos de administrador.
- Não revisar as permissões para entender se elas permitem o acesso de privilégio mínimo.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientações para a implementação

O princípio de estados [privilégio mínimo](#) para as identidades deve ser apenas permitido para realizar o mínimo de ações necessárias para cumprir uma tarefa específica. Isso equilibra a usabilidade, eficiência e segurança. Operar sobre esse princípio ajuda a limitar acesso não intencional e a rastrear quem tem acesso a quais recursos. Usuários e perfis do IAM não têm permissões por padrão. O usuário raiz tem acesso total e deve ser controlado e monitorado rigidamente, além de usado apenas para [tarefas que necessitam acesso raiz](#).

Políticas do IAM são usadas para conceder explicitamente permissões aos perfis do IAM ou recursos específicos. Por exemplo, políticas com base em identidade podem ser anexadas a grupos do IAM, enquanto buckets do S3 podem ser controlados por políticas baseadas em recursos.

Ao criar e associar uma política do IAM, você pode especificar as ações de serviço, os recursos e as condições que devem ser verdadeiras para que a AWS permita ou negue o acesso. A AWS oferece suporte a uma variedade de condições para ajudar você a reduzir o acesso. Por exemplo, ao usar `PrincipalOrgID` como [chave de condição](#), você pode negar ações se o solicitante não for parte da sua organização da AWS.

Você também pode controlar as solicitações feitas pelos serviços da AWS em seu nome, como a criação, pelo AWS CloudFormation, de uma função do AWS Lambda, usando a chave de condição `CalledVia`. Tipos diferentes de política devem estar em camadas para estabelecer a defesa em profundidade e limitar as permissões gerais de seus usuários. Você pode restringir as permissões que podem ser concedidas e sob quais condições. Por exemplo, você pode permitir que suas equipes de aplicação criem suas próprias políticas do IAM para os sistemas que criam, mas deve também aplicar uma [Fronteira de permissão](#) para limitar o máximo de permissões que o sistema pode receber.

Etapas da implementação

- Implementar políticas de privilégio mínimo: atribua políticas de acesso com privilégio mínimo a grupos e perfis do IAM para refletir a função ou o perfil do usuário que você definiu.
- Basear as políticas no uso da API: uma maneira de determinar as permissões necessárias é analisar os logs do AWS CloudTrail. Essa análise permite que você crie permissões personalizadas para as ações do usuário dentro da AWS. O [IAM Access Analyzer pode gerar automaticamente uma IAM política com base na atividade](#). Você pode usar o IAM Access Advisor no nível da organização ou da conta para [rastrear as últimas informações acessadas para determinada política](#).

- Considerar o uso de [políticas gerenciadas da AWS para cargos](#). Pode ser difícil saber por onde começar ao criar políticas de permissões mais estritas. A AWS gerencia políticas para cargos comuns, como faturamento, administradores de banco de dados e cientistas de dados. Essas políticas podem ajudar a diminuir o acesso dos usuários ao determinar como implementar as políticas de privilégio mínimo.
- Remover permissões desnecessárias: remova permissões que não são necessárias e ajuste políticas muito permissivas. A [geração de política pelo IAM Access Analyzer](#) pode ajudar a ajustar as políticas de permissão.
- Garantir que os usuários tenham acesso limitado a ambientes de produção: os usuários devem ter acesso a ambientes de produção apenas com um caso de uso válido. Depois de o usuário realizar as tarefas específicas para as quais foi necessário o acesso à produção, o acesso deve ser revogado. Limitar o acesso a ambientes de produção evita eventos não intencionais e que causam impacto à produção, além de diminuir o escopo do impacto do acesso não intencional.
- Considerar os limites de permissões: um limite de permissões é um recurso para usar uma política gerenciada que define o número máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM. O limite de permissões de uma entidade permite que ela execute apenas as ações aceitas por suas políticas baseadas em identidade e seus limites de permissões.
- Considerar [tags de recursos](#) para permissões: um modelo de controle de acesso baseado em atributo que usa tags de recursos permite conceder acesso com base no propósito do recurso, proprietário, ambiente ou outros critérios. Por exemplo, você pode usar tags de recurso para diferenciar entre ambientes de desenvolvimento e produção. Ao usar essas tags, é possível restringir os desenvolvedores ao ambiente de desenvolvimento. Ao combinar as tags e as políticas de permissões, você consegue alcançar um acesso restrito ao recurso sem precisar definir políticas complicadas e personalizadas para cada cargo.
- Use [políticas de controle de serviço](#) para AWS Organizations. As políticas de controle de serviço controlam centralmente o máximo de permissões disponíveis para contas de membros em sua organização. É importante notar que as políticas de controle de serviço permitem que você restrinja as permissões do usuário raiz nas contas de membros. Considere também o uso do AWS Control Tower, que fornece controles gerenciados prescritivos que enriquecem o AWS Organizations. Também é possível definir os seus próprios controles no Control Tower.
- Estabelecer uma política de ciclo de vida para sua organização: as políticas de ciclo de vida do usuário definem tarefas a serem realizadas quando os usuários entram na AWS, mudam de cargo ou escopo de trabalho ou não precisam mais de acesso à AWS. As análises de permissão devem

ser feitas durante todas as etapas do ciclo de vida do usuário para verificar se as permissões estão adequadamente restritas e para evitar desvios nas permissões.

- Estabelecer uma programação regular para rever as permissões e remover as permissões desnecessárias: frequentemente, você deve verificar o acesso do usuário para garantir que ele não tenha acesso muito permissivo. O [AWS Config](#) e o IAM Access Analyzer podem ajudar ao auditar as permissões do usuário.
- Estabelecer uma matriz de cargos: uma matriz de cargos exibe os diversos cargos e níveis de acesso necessários dentro de sua área da AWS. Com uma matriz de cargos, você pode definir e separar as permissões com base nas responsabilidades do usuário dentro da sua organização. Use grupos em vez de aplicar permissões diretamente a usuários ou cargos individuais.

Recursos

Documentos relacionados:

- [Conceder privilégio mínimo](#)
- [Permissions boundaries for IAM entities](#) (Limites de permissões para entidades do IAM)
- [Techniques for writing least privilege IAM policies](#) (Técnicas para escrever políticas do IAM de privilégio mínimo)
- [IAM Access Analyzer makes it easier to implement least privilege permissions by generating IAM policies based on access activity](#) (IAM Access Analyzer facilita a implementação de permissões de privilégio mínimo gerando políticas do IAM baseadas na atividade de acesso)
- [Delegate permission management to developers by using IAM permissions boundaries](#) (Delegar gerenciamento de permissões para desenvolvedores usando os limites de permissões do IAM)
- [Refining Permissions using last accessed information \(Refinar permissões usando as últimas informações acessadas\)](#)
- [IAM policy types and when to use them](#) (Tipos de política do IAM e quando usá-las)
- [Testing IAM policies with the IAM policy simulator](#) (Testar políticas do IAM com o simulador de política do IAM)
- [Guardrails in AWS Control Tower](#) (Barreiras de proteção no AWS Control Tower)
- [Zero Trust architectures: An AWS perspective](#) (Arquiteturas de confiança zero: uma perspectiva da AWS)
- [How to implement the principle of least privilege with CloudFormation StackSets](#) (Como implementar o princípio de privilégio mínimo com o CloudFormation StackSets)

- [Controle de acesso baseado em atributos \(ABAC\)](#)
- [Redução do escopo da política ao exibir a atividade do usuário](#)
- [Visualizar acesso do cargo](#)
- [Use a marcação para organizar seu ambiente e gerar responsabilidade](#)
- [Estratégias de marcação da AWS](#)
- [Marcação de recursos da AWS](#)

Vídeos relacionados:

- [Next-generation permissions management \(Gerenciamento de permissões de última geração\)](#)
- [Zero Trust: An AWS perspective](#) (Confiança zero: uma perspectiva da AWS)
- [How can I use permissions boundaries to limit users and roles to prevent privilege escalation?](#) (Como posso usar limites de permissões para limitar usuários e funções e evitar escalção do privilégio?)

Exemplos relacionados:

- [Lab: IAM permissions boundaries delegating role creation](#) (Laboratório: limites de permissões do IAM que delegam a criação de perfis)
- [Lab: IAM tag based access control for EC2](#) (Laboratório: controle de acesso baseado em tags do IAM para EC2)

SEC03-BP03 Estabelecer processo de acesso de emergência

Crie um processo que permita acesso emergencial às suas workloads no caso improvável de um problema com seu provedor de identidades centralizado.

Você deve criar processos para diferentes modos de falha que possam resultar em um evento de emergência. Por exemplo, em circunstâncias normais, os usuários da sua força de trabalho são federados na nuvem usando um provedor de identidades centralizado ([SEC02-BP04](#)) para gerenciar as respectivas workloads. No entanto, se o provedor de identidades centralizado falhar ou a configuração da federação na nuvem for modificada, talvez os usuários de sua força de trabalho não consigam se federar na nuvem. Um processo de acesso de emergência permite que administradores autorizados acessem seus recursos de nuvem por meios alternativos (como uma forma alternativa de federação ou acesso direto do usuário) para corrigir problemas com sua

configuração de federação ou workloads. O processo de acesso de emergência é usado até que o mecanismo normal de federação seja restaurado.

Resultado desejado:

- Você definiu e documentou os modos de falha que são considerados uma emergência: considere suas circunstâncias normais e os sistemas dos quais seus usuários dependem para gerenciar suas workloads. Pense em como cada uma dessas dependências pode falhar e causar uma situação de emergência. Você pode encontrar as perguntas e as práticas recomendadas no [Pilar Confiabilidade](#) útil para identificar modos de falha e arquitetar sistemas mais resilientes com o objetivo de minimizar a probabilidade de falhas.
- Você documentou as etapas que devem ser seguidas para confirmar uma falha como emergência. Por exemplo, é possível exigir que os administradores de identidade confirmem o status de seus provedores de identidade primário e de reserva e, se nenhum dos dois estiver disponível, declarar um evento de emergência por falha do provedor de identidades.
- Você definiu um processo de acesso de emergência específico de cada tipo de modo de emergência ou falha. Ser específico pode reduzir a tentação de seus usuários de abusar de um processo geral para todos os tipos de emergência. Seus processos de acesso de emergência descrevem as circunstâncias em que cada processo deve ser usado e, inversamente, as situações em que o processo não deve ser usado e apontam para processos alternativos que podem ser aplicados.
- Seus processos são bem documentados com instruções detalhadas e manuais que podem ser seguidos com rapidez e eficiência. Lembre-se de que um evento de emergência pode ser um momento estressante para os usuários e eles podem estar sob extrema pressão de tempo, portanto, projete o processo para ser o mais simples possível.

Antipadrões comuns:

- Você não tem processos de acesso de emergência bem documentados e bem testados. Os usuários não estão preparados para uma emergência e seguem processos improvisados quando surge um evento de emergência.
- Seus processos de acesso de emergência dependem dos mesmos sistemas (como um provedor de identidades centralizado) que seus mecanismos de acesso normais. Isso significa que a falha desse sistema pode afetar os mecanismos de acesso normal e de emergência e prejudicar sua capacidade de se recuperar da falha.

- Seus processos de acesso de emergência são usados em situações não emergenciais. Por exemplo, os usuários frequentemente usam de forma indevida os processos de acesso de emergência, pois acham mais fácil fazer alterações diretamente do que enviá-las por meio de um pipeline.
- Seus processos de acesso de emergência não geram logs suficientes para auditar os processos, ou os logs não são monitorados para alertar sobre o possível uso indevido dos processos.

Benefícios de estabelecer esta prática recomendada:

- Com processos de acesso de emergência bem documentados e testados, é possível reduzir o tempo gasto pelos usuários para responder e resolver um evento de emergência. Isso pode resultar em menos tempo de inatividade e maior disponibilidade dos serviços fornecidos aos seus clientes.
- Você pode rastrear cada solicitação de acesso de emergência e detectar e alertar sobre tentativas não autorizadas de uso indevido do processo para eventos não emergenciais.

Nível de risco exposto se esta prática recomendada não for estabelecida: Médio

Orientação para implementação

Esta seção fornece orientação para criar processos de acesso de emergência para vários modos de falha relacionados às workloads implantadas na AWS, começando com uma orientação comum que se aplica a todos os modos de falha e seguida por uma orientação específica com base no tipo de modo de falha.

Orientação comum para todos os modos de falha

Pense no seguinte ao projetar um processo de acesso de emergência para um modo de falha:

- Documente as pré-condições e as suposições do processo: quando o processo deve ou não ser usado. Isso ajuda a detalhar o modo de falha e documentar suposições, como o estado de outros sistemas relacionados. Por exemplo, o processo do Modo de falha 2 pressupõe que o provedor de identidades está disponível, mas a configuração na AWS foi modificada ou expirou.
- Pré-crie os recursos necessários para o processo de acesso de emergência ([SEC10-BP05](#)). Por exemplo, crie previamente a Conta da AWS de acesso de emergência com IAM users e perfis e os perfis entre contas do IAM em todas as contas da workload. Isso verifica se esses recursos estão prontos e disponíveis quando ocorre um evento de emergência. Ao pré-criar recursos, você não depende das APIs do ambiente de gerenciamento da AWS ([usadas para criar e modificar recursos](#))

da AWS) que podem ficar indisponíveis em caso de emergência. Além disso, ao pré-criar recursos do IAM, você não precisa contabilizar [possíveis atrasos devido à eventual consistência](#).

- Inclua processos de acesso de emergência como parte dos planos de gerenciamento de incidentes ([SEC10-BP02](#)). Documente como os eventos de emergência são acompanhados e comunicados a outras pessoas na organização, como equipes de colegas, sua liderança e, quando aplicável, externamente a seus clientes e parceiros de negócios.
- Defina o processo de solicitação de acesso de emergência no sistema de fluxo de trabalho de solicitação de serviço existente, caso haja um. Normalmente, esses sistemas de fluxo de trabalho permitem criar formulários de admissão para coletar informações sobre a solicitação, acompanhar a solicitação em cada estágio do fluxo de trabalho e adicionar etapas de aprovação automatizadas e manuais. Relacione cada solicitação a um evento de emergência correspondente acompanhado no sistema de gerenciamento de incidentes. Ter um sistema uniforme para acessos de emergência permite que você acompanhe essas solicitações em um único sistema, analise as tendências de uso e melhore os processos.
- Verifique se os processos de acesso de emergência só podem ser iniciados por usuários autorizados e exigem aprovações dos colegas ou da gerência do usuário, conforme apropriado. O processo de aprovação deve operar de forma eficaz dentro e fora do horário comercial. Defina como as solicitações de aprovação permitirão aprovadores secundários se os aprovadores primários não estiverem disponíveis e forem encaminhadas para a cadeia de gerenciamento até serem aprovadas.
- Verifique se o processo gera logs e eventos de auditoria detalhados para tentativas bem-sucedidas e fracassadas de obter acesso de emergência. Monitore o processo de solicitação e o mecanismo de acesso de emergência para detectar uso indevido ou acessos não autorizados. Correlacione a atividade com eventos de emergência contínuos do sistema de gerenciamento de incidentes e alerte quando as ações ocorrerem fora dos períodos esperados. Por exemplo, você deve monitorar e alertar sobre atividades na Conta da AWS de acesso de emergência, pois ela nunca deve ser usada em operações normais.
- Teste os processos de acesso de emergência periodicamente para verificar se as etapas estão claras e garantir o nível correto de acesso com rapidez e eficiência. Os processos de acesso de emergência devem ser testados como parte das simulações de resposta a incidentes ([SEC10-BP07](#)) e testes de recuperação de desastres ([REL13-BP03](#)).

Modo de falha 1: o provedor de identidades usado para federar na AWS não está disponível

Conforme descrito em [SEC02-BP04 Contar com um provedor de identidades centralizado](#), recomendamos confiar em um provedor de identidades centralizado para federar os usuários de

sua força de trabalho e conceder acesso a Contas da AWS. Você pode federar em várias Contas da AWS na organização da AWS usando o IAM Identity Center ou federar em Contas da AWS individuais usando o IAM. Nos dois casos, os usuários da força de trabalho se autenticam com seu provedor de identidades centralizado antes de serem redirecionados a um endpoint de login da AWS para SSO.

No caso improvável do provedor de identidades centralizado não estar disponível, os usuários da sua força de trabalho não poderão se federar nas Contas da AWS nem gerenciar as workloads. Nesse evento de emergência, é possível fornecer um processo de acesso de emergência para um pequeno grupo de administradores acessar Contas da AWS a fim de realizar tarefas essenciais que não podem esperar até que seus provedores de identidades centralizados estejam online novamente. Por exemplo, seu provedor de identidades fica indisponível por quatro horas e, durante esse período, você precisa modificar os limites superiores de um grupo do Amazon EC2 Auto Scaling em uma conta de produção para lidar com um aumento inesperado no tráfego de clientes. Seus administradores de emergência devem seguir o processo de acesso de emergência a fim de obter acesso à Conta da AWS de produção específica e fazer as alterações necessárias.

O processo de acesso de emergência depende de uma Conta da AWS de acesso de emergência pré-criada usada exclusivamente para acesso de emergência e tem recursos da AWS (como perfis do IAM e IAM users) para apoiar o processo de acesso de emergência. Durante as operações normais, ninguém deve acessar a conta de acesso de emergência, e você deve monitorar e alertar sobre o uso indevido dessa conta (para receber mais detalhes, consulte a seção [Orientação comum anterior](#)).

A conta de acesso de emergência tem perfis do IAM de acesso de emergência com permissões para assumir perfis entre contas nas Contas da AWS que exigem acesso de emergência. Esses perfis do IAM são pré-criados e configurados com políticas de confiança que confiam nos perfis do IAM da conta de emergência.

O processo de acesso de emergência pode usar uma das seguintes abordagens:

- Você pode pré-criar um conjunto de [IAM users](#) para seus administradores de emergência na conta de acesso de emergência com senhas fortes e tokens de MFA associados. Esses IAM users têm permissões para assumir os perfis do IAM que permitem o acesso entre contas à Conta da AWS onde o acesso de emergência é necessário. Recomendamos criar o menor número possível de usuários e atribuir cada um a um único administrador de emergência. Durante uma emergência, um usuário administrador de emergência entra na conta de acesso de emergência usando sua senha e código de token MFA, muda para a função do IAM de acesso de emergência na conta de emergência e, por fim, muda para a função do IAM de acesso de emergência na conta da

workload para realizar a ação de alteração de emergência. A vantagem dessa abordagem é que cada IAM user é atribuído a um administrador de emergência, e você pode saber qual usuário fez login analisando os eventos do CloudTrail. A desvantagem é que você precisa manter vários IAM users com as respectivas senhas de longa duração e tokens de MFA associados.

- Você pode usar o [usuário raiz da Conta da AWS de acesso de emergência](#) para entrar na conta de acesso de emergência, assumir o perfil do IAM para acesso de emergência e assumir o perfil entre contas na conta da workload. Recomendamos definir uma senha forte e vários tokens de MFA para o usuário raiz. Também recomendamos armazenar a senha e os tokens de MFA em um cofre de credenciais corporativo seguro que imponha autenticação e autorização fortes. Você deve proteger a senha e os fatores de redefinição de tokens de MFA: defina o endereço de e-mail da conta como uma lista de distribuição de e-mail monitorada pelos administradores de segurança na nuvem e o número de telefone da conta como um número de telefone compartilhado que também seja monitorado pelos administradores de segurança. A vantagem dessa abordagem é que há um conjunto de credenciais de usuário raiz para gerenciar. A desvantagem é que, como se trata de um usuário compartilhado, vários administradores podem fazer login como usuário raiz. Você deve fazer auditoria dos eventos de log do cofre corporativo para identificar qual administrador fez check-out da senha do usuário raiz.

Modo de falha 2: a configuração do provedor de identidades na AWS foi modificada ou expirou

Para permitir que os usuários de sua força de trabalho sejam federados nas Contas da AWS, você pode configurar o IAM Identity Center com um provedor de identidades externo ou criar um provedor de identidades do IAM ([SEC02-BP04](#)). Normalmente, você os configura importando um documento XML de metadados SAML fornecido pelo provedor de identidades. O documento XML de metadados inclui um certificado X.509 correspondente a uma chave privada que o provedor de identidades usa para assinar as declarações SAML.

Essas configurações no lado da AWS podem ser modificadas ou excluídas por engano por um administrador. Em outro cenário, o certificado X.509 importado para a AWS pode expirar, e um novo XML de metadados com um novo certificado ainda não foi importado para a AWS. Os dois cenários podem interromper a federação na AWS para os usuários de sua força de trabalho, ocasionando uma emergência.

Nesse evento de emergência, você pode fornecer aos seus administradores de identidade acesso à AWS para resolver os problemas de federação. Por exemplo, seu administrador de identidade usa o processo de acesso de emergência para fazer login na Conta da AWS de acesso de emergência, muda para um perfil na conta de administrador do Centro de Identidade e atualiza a configuração do

provedor de identidades externo importando o documento XML de metadados SAML mais recente do provedor de identidades para reativar a federação. Depois que a federação for corrigida, os usuários da sua força de trabalho continuarão usando o processo operacional normal para federar em suas contas da workload.

Você pode seguir as abordagens detalhadas no Modo de falha 1 anterior para criar um processo de acesso de emergência. É possível conceder permissões de privilégio mínimo aos seus administradores de identidade a fim de acessar somente a conta de administrador do Centro de Identidade e realizar ações no Centro de Identidade nessa conta.

Modo de falha 3: interrupção do Centro de Identidade

No caso improvável de uma interrupção do IAM Identity Center ou da Região da AWS, recomendamos definir uma configuração que possa ser usada para conceder acesso temporário ao AWS Management Console.

O processo de acesso de emergência usa a federação direta do provedor de identidades no IAM em uma conta de emergência. Para receber detalhes sobre as considerações sobre o processo e o design, consulte [Configurar o acesso de emergência ao AWS Management Console](#).

Etapas da implementação

Etapas comuns para todos os modos de falha

- Crie uma Conta da AWS dedicado aos processos de acesso de emergência. Pré-crie os recursos do IAM necessários na conta, como perfis do IAM ou IAM users e, opcionalmente, provedores de identidades do IAM. Além disso, crie previamente perfis do IAM entre contas nas Contas da AWS da workload com relacionamentos de confiança com os perfis do IAM correspondentes na conta de acesso de emergência. Você pode usar o [AWS CloudFormation StackSets com AWS Organizations](#) para criar esses recursos nas contas de membros de sua organização.
- Crie políticas de controle de serviço do AWS Organizations ([SCPS](#)) para negar a exclusão e a modificação dos perfis do IAM entre contas nas Contas da AWS de membros.
- Ative o CloudTrail para a Conta da AWS de acesso de emergência e envie os eventos da trilha a um bucket central do S3 em sua Conta da AWS de coleção de logs. Se você estiver usando o AWS Control Tower para configurar e controlar seu ambiente de várias contas da AWS, todas as contas que você criar usando o AWS Control Tower ou inscrever no AWS Control Tower terão o CloudTrail ativado por padrão e serão enviadas a um bucket do S3 em uma Conta da AWS de arquivo de log dedicado.

- Monitore a atividade da conta de acesso de emergência criando regras do EventBridge que correspondam ao login do console e à atividade da API pelos perfis de emergência do IAM. Envie notificações ao seu centro de operações de segurança quando ocorrerem atividades fora de um evento de emergência contínuo acompanhado no sistema de gerenciamento de incidentes.

Etapas adicionais para o Modo de falha 1: o provedor de identidades usado para federar na AWS não está disponível, e o Modo de falha 2: a configuração do provedor de identidades na AWS foi modificada ou expirou

- Pré-crie recursos de acordo com o mecanismo escolhido para acesso de emergência:
 - Usar IAM users: pré-crie-os IAM users com senhas fortes e dispositivos de MFA associados.
 - Usar o usuário raiz da conta de emergência: configure o usuário raiz com uma senha forte e armazene a senha no seu cofre de credenciais corporativo. Associe vários dispositivos físicos de MFA ao usuário raiz e armazene os dispositivos em locais que possam ser acessados rapidamente pelos membros de sua equipe de administradores de emergência.

Etapas adicionais para o Modo de falha 3: interrupção do Centro de Identidade

- Conforme detalhado em [Configurar o acesso de emergência ao AWS Management Console](#), na Conta da AWS de acesso de emergência, crie um provedor de identidades do IAM para ativar a federação direta de SAML a partir do provedor de identidades.
- Crie grupos de operações de emergência no IdP sem membros.
- Crie perfis do IAM correspondentes aos grupos de operações de emergência na conta de acesso de emergência.

Recursos

Práticas recomendadas relacionadas ao Well-Architected:

- [SEC02-BP04 Contar com um provedor de identidades centralizado](#)
- [SEC03-BP02 Conceder acesso com privilégio mínimo](#)
- [SEC10-BP02 Desenvolver planos de gerenciamento de incidentes](#)
- [SEC10-BP07 Promover dias de jogo](#)

Documentos relacionados:

- [Configurar o acesso de emergência ao AWS Management Console](#)
- [Permitir que usuários federados do SAML 2.0 acessem o AWS Management Console](#)
- [Acesso de emergência](#)

Vídeos relacionados:

- [AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center](#)
- [AWS re:Inforce 2022 - AWS Identity and Access Management \(IAM\) deep dive](#)

Exemplos relacionados:

- [Perfil de acesso de emergência da AWS](#)
- [AWS customer playbook framework](#)
- [AWS incident response playbook samples](#)

SEC03-BP04 Reduzir as permissões continuamente

À medida que suas equipes determinarem o acesso de que precisam, remova as permissões desnecessárias e estabeleça processos de análise para obter permissões de privilégio mínimo. Monitore e remova continuamente identidades e permissões não utilizadas para acesso humano e de máquina.

Resultado desejado: as políticas de permissão devem seguir o princípio de privilégio mínimo. À medida que os cargos e os perfis se tornem mais bem definidos, suas políticas de permissões precisam ser analisadas para remover permissões desnecessárias. Essa abordagem reduz o escopo do impacto caso as credenciais sejam expostas de forma acidental ou sejam acessadas sem autorização.

Antipadrões comuns:

- Usar como padrão a concessão de permissões de administrador aos usuários.
- Criar políticas permissivas demais, mas sem privilégios completos de administrador.
- Manter as políticas de permissão quando não são mais necessárias.

Nível de risco exposto se esta prática recomendada não é estabelecida: médio

Orientação de implementação

Enquanto as equipes e os projetos estiverem começando, políticas de permissão permissivas podem ser usadas para inspirar inovação e agilidade. Por exemplo, em um ambiente de desenvolvimento ou teste, os desenvolvedores podem receber acesso a uma ampla gama de serviços da AWS. Recomendamos avaliar o acesso de forma contínua e restringir o acesso somente àqueles serviços e ações de serviço necessários para concluir o trabalho atual. Recomendamos essa avaliação para identidades humanas e de máquina. Identidades de máquina, às vezes, denominadas contas de sistema ou serviço, são identidades que fornecem acesso da AWS a aplicações ou servidores. Esse acesso é especialmente importante em um ambiente de produção, em que as permissões excessivamente permissivas podem causar um grande impacto e expor dados dos clientes.

A AWS oferece vários métodos para ajudar a identificar usuários, perfis, permissões e credenciais não utilizados. A AWS também pode ajudar a analisar a atividade de acesso dos usuários e dos perfis do IAM, como chaves de acesso associadas, e o acesso aos recursos da AWS, como objetos em buckets do Amazon S3. A geração de políticas do AWS Identity and Access Management Access Analyzer pode auxiliar você a criar políticas de permissão restritivas com base nos serviços e nas ações reais com os quais uma entidade principal interage. [O controle de acesso baseado em atributo \(ABAC\)](#) pode ajudar a simplificar o gerenciamento de permissões, pois você pode conceder permissões aos usuários utilizando os atributos deles em vez de anexar políticas de permissões diretamente a cada usuário.

Etapas da implementação

- Utilizar o [AWS Identity and Access Management Access Analyzer](#): o IAM Access Analyzer ajuda a identificar os recursos na organização e nas contas, como buckets do Amazon Simple Storage Service (Amazon S3) ou perfis do IAM, que são [compartilhados com uma entidade externa](#).
- Utilizar a [geração de políticas do IAM Access Analyzer](#): a geração de políticas do IAM Access Analyzer ajuda você a [criar políticas de permissão detalhadas com base em um usuário do IAM ou na atividade de acesso de um perfil](#).
- Determinar um cronograma e uma política de uso aceitáveis para usuários e perfis do IAM: utilize o [carimbo de data e hora de último acesso](#) para [identificar usuários e perfis não utilizados](#) e removê-los. Revise as informações de serviço e ação acessadas mais recentemente para identificar e [definir o escopo das permissões para usuários e perfis específicos](#). Por exemplo, você pode usar as informações acessadas mais recentemente para identificar as ações específicas do Amazon S3 exigidas pelo perfil da aplicação e restringir o acesso do perfil apenas a essas ações. Os recursos de informações acessadas mais recentemente estão disponíveis no AWS Management

Console e de maneira programática para permitir que você os incorpore aos fluxos de trabalho de infraestrutura e ferramentas automatizadas.

- Considerar [o registro em log dos eventos de dados no AWS CloudTrail](#): por padrão, o CloudTrail não registra eventos de dados, como atividade em nível de objeto do Amazon S3 (por exemplo, GetObject e DeleteObject) ou atividades de tabelas do Amazon DynamoDB (por exemplo, PutItem e DeleteItem). Considere ativar o registro em log desses eventos para determinar quais usuários e perfis precisam acessar objetos do Amazon S3 ou itens de tabelas do DynamoDB específicos.

Recursos

Documentos relacionados:

- [Conceder privilégio mínimo](#)
- [Remova credenciais desnecessárias](#)
- [O que é o AWS CloudTrail?](#)
- [Trabalhando com políticas](#)
- [Registrar em log e monitorar no DynamoDB](#)
- [Habilitar o log de eventos do CloudTrail para buckets e objetos do Amazon S3](#)
- [Obter relatórios de credenciais da sua Conta da AWS](#)

Vídeos relacionados:

- [Torne-se um mestre em políticas do IAM em 60 minutos ou menos](#)
- [Separação de tarefas, privilégio mínimo, delegação e CI/CD](#)
- [AWS re:Inforce 2022: Aprofundamento no AWS Identity and Access Management \(IAM\)](#)

SEC03-BP05 Definir barreiras de proteção de permissões para sua organização

Use barreiras de proteção de permissões para reduzir o escopo das permissões disponíveis que podem ser concedidas a entidades principais. A cadeia de avaliação de políticas de permissão inclui barreiras de proteção para determinar as permissões efetivas de uma entidade principal ao tomar decisões de autorização. Você pode definir barreiras de proteção usando uma abordagem baseada

em camadas. Aplique algumas barreiras de proteção de maneira abrangente em toda a organização e outras de forma detalhada às sessões de acesso temporário.

Resultado desejado: ter um nítido isolamento dos ambientes usando Contas da AWS separadas. As políticas de controle de serviços (SCPs) são usadas para definir barreiras de proteção de permissões em toda a organização. As barreiras de proteção mais amplas são definidas nos níveis hierárquicos mais próximos da raiz da sua organização e as barreiras de proteção mais rígidas são definidas mais perto do nível das contas individuais. Quando admitidas, as políticas de recursos definem as condições que uma entidade principal deve satisfazer para obter acesso a um recurso. As políticas de recursos também abrangem o conjunto de ações permitidas, quando apropriado. Os limites de permissão são impostos às entidades principais que gerenciam as permissões da workload, delegando o gerenciamento de permissões aos proprietários de workload individuais.

Antipadrões comuns:

- Criar Contas da AWS de membros dentro de uma [organização da AWS](#), mas não usar SCPs para restringir o uso e as permissões disponíveis para as respectivas credenciais-raiz.
- Atribuir permissões com base em privilégio mínimo, mas não colocar barreiras de proteção no conjunto máximo de permissões que podem ser concedidas.
- Confiar no princípio de negação implícita do AWS IAM para restringir permissões, acreditando que essas políticas não concederão uma permissão explícita indesejada.
- Executar vários ambientes de workload na mesma Conta da AWS e confiar em mecanismos como VPCs, tags ou políticas de recursos para impor limites de permissão.

Benefícios do estabelecimento desta prática recomendada: as barreiras de proteção de permissões ajudam a criar a confiança de que não serão concedidas permissões indesejadas mesmo quando uma política de permissão tentar fazê-lo. Isso pode simplificar a definição e o gerenciamento de permissões, reduzindo o escopo máximo das permissões que precisam ser consideradas.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Recomendamos que você use uma abordagem baseada em camadas para definir barreiras de proteção de permissões para sua organização. Essa abordagem reduz sistematicamente o conjunto máximo de permissões possíveis à medida que camadas adicionais são aplicadas. Isso ajuda você a conceder acesso com base no princípio de privilégio mínimo, reduzindo o risco de acesso indesejado devido à configuração incorreta da política.

A primeira etapa para estabelecer barreiras de proteção de permissões é isolar workloads e ambientes em Contas da AWS separadas. As entidades principais de uma conta não podem acessar recursos em outra conta sem permissão explícita para isso, mesmo quando as duas contas estão na mesma organização da AWS ou na mesma [unidade organizacional \(UO\)](#). Você pode usar UOs para agrupar contas que deseja administrar como uma única unidade.

A próxima etapa é reduzir o conjunto máximo de permissões que você pode conceder às entidades principais nas contas de membros da sua organização. Você pode usar [políticas de controle de serviços \(SCPs\)](#) para essa finalidade, as quais podem ser aplicadas a uma UO ou a uma conta. As SCPs podem impor controles de acesso comuns, como restringir o acesso a determinadas Regiões da AWS, ajudar a impedir a exclusão de recursos ou desabilitar ações de serviço possivelmente arriscadas. As SCPs que você aplica à raiz da sua organização afetam apenas as contas dos membros, não a conta de gerenciamento. As SCPs regem apenas as entidades principais de sua organização. As SCPs não regem entidades principais externas à sua organização que estão acessando seus recursos.

Outra etapa é usar [políticas de recursos do IAM](#) para definir o escopo das ações disponíveis que você pode realizar nos recursos que elas governam, bem como quaisquer condições que a entidade principal interina deva atender. Isso pode ser tão amplo quanto permitir todas as ações, desde que a entidade principal faça parte de sua organização (usando a [chave de condição](#) PrincipalOrgId), ou tão detalhado quanto permitir apenas ações específicas por parte de um perfil específico do IAM. Você pode adotar uma abordagem semelhante com as condições das políticas de confiança de perfis do IAM. Se uma política de confiança de recursos ou perfis nomear explicitamente uma entidade principal na mesma conta que o perfil ou o recurso que ela governa, essa entidade principal não precisará de uma política do IAM anexada que conceda as mesmas permissões. Se a entidade principal estiver em uma conta diferente da conta do recurso, ela precisará de uma política do IAM anexada que conceda essas permissões.

Muitas vezes, uma equipe de workload quer gerenciar as permissões que a respectiva workload exige. Isso, por sua vez, pode exigir que a equipe crie políticas de permissão e perfis do IAM.

Você pode capturar o escopo máximo de permissões que a equipe pode conceder em um [limite de permissão do IAM](#) e associar esse documento a um perfil do IAM que a equipe possa usar para gerenciar os respectivos perfis e permissões do IAM. Essa abordagem pode dar liberdade à equipe para concluir seu trabalho e, ao mesmo tempo, reduzir os riscos de acesso administrativo ao IAM.

Uma etapa mais detalhada é implementar técnicas de gerenciamento de acesso privilegiado (PAM) e gerenciamento temporário de acesso elevado (TEAM). Um exemplo de PAM é exigir que as entidades principais realizem a autenticação multifator antes de executar ações privilegiadas. Para

obter mais informações, consulte [Configuração de acesso à API protegido por MFA](#). O TEAM exige uma solução que gerencie a aprovação e o período durante o qual uma entidade principal pode ter acesso elevado. Uma abordagem é adicionar temporariamente a entidade principal à política de confiança de perfis referente a um perfil do IAM que tenha acesso elevado. Outra abordagem, em operação normal, é reduzir as permissões concedidas a uma entidade principal por um perfil do IAM usando uma [política de sessão](#) e, em seguida, suspender temporariamente essa restrição durante o período aprovado. Para saber mais sobre soluções validadas da AWS e de alguns parceiros, consulte [Temporary elevated access](#).

Etapas da implementação

1. Isole workloads e ambientes em Contas da AWS separadas.
2. Use SCPs para reduzir o conjunto máximo de permissões que podem ser concedidas às entidades principais nas contas dos membros da sua organização.
 - a. Recomendamos que você adote uma abordagem de lista de permissões para redigir as SCPs que negam todas as ações, exceto aquelas permitidas, bem como descrever as condições sob as quais elas são permitidas. Defina primeiro os recursos que você deseja controlar e, em seguida, o efeito de negar. Use o elemento NotAction para negar todas as ações, exceto aquelas que você especificar. Associe isso a uma condição NotLike para definir quando essas ações são permitidas, se aplicável, como StringNotLike e ArnNotLike.
 - b. Consulte [Service control policy examples](#).
3. Use políticas de recursos do IAM para reduzir o escopo e especificar as condições das ações permitidas nos recursos. Use as condições nas políticas de confiança de perfis do IAM para criar restrições aos perfis assumidos.
4. Atribua limites de permissão do IAM aos perfis do IAM que as equipes de workload podem usar para gerenciar perfis e permissões do IAM de suas próprias workloads.
5. Avalie as soluções PAM e TEAM com base em suas necessidades.

Recursos

Documentos relacionados:

- [Data perimeters on AWS](#)
- [Establish permissions guardrails using data perimeters](#)
- [Policy evaluation logic](#)

Exemplos relacionados:

- [Service control policy examples](#)

Ferramentas relacionadas:

- [Solução da AWS: Temporary Elevated Access Management](#)
- [Validated security partner solutions for TEAM](#)

SEC03-BP06 Gerenciar o acesso com base no ciclo de vida

Monitore e ajuste as permissões concedidas às entidades principais (usuários, funções e grupos) durante todo o ciclo de vida em sua organização. Ajuste as associações de grupo à medida que os usuários mudarem de função e remova o acesso quando um usuário sair da organização.

Resultado desejado: as permissões são monitoradas e ajustadas em todo o ciclo de vida das entidades principais da organização, reduzindo o risco de privilégios desnecessários. Você concede acesso apropriado ao criar um usuário. Você modifica o acesso à medida que as responsabilidades do usuário mudam e remove o acesso quando o usuário não está mais ativo ou sai da organização. Você gerencia centralmente as alterações em seus usuários, funções e grupos. Você usa a automação para propagar alterações em seus ambientes da AWS.

Antipadrões comuns:

- Você concede antecipadamente privilégios de acesso excessivos ou amplos às identidades, além daqueles exigidos inicialmente.
- Você não revisa e ajusta os privilégios de acesso à medida que as funções e responsabilidades das identidades mudam ao longo do tempo.
- Você mantém identidades inativas ou encerradas com privilégios de acesso ativos. Isso aumenta o risco de acesso não autorizado.
- Você não automatiza o gerenciamento dos ciclos de vida da identidade.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Gerencie e ajuste cuidadosamente os privilégios de acesso que você concede às identidades (como usuários, funções, grupos) durante todo o ciclo de vida. Esse ciclo de vida inclui a fase inicial de

integração, mudanças contínuas nas funções e responsabilidades e eventual desligamento ou rescisão. Gerencie proativamente o acesso com base no estágio do ciclo de vida para manter o nível de acesso adequado. Siga o princípio do privilégio mínimo para reduzir o risco de privilégios de acesso excessivos ou desnecessários.

Você pode gerenciar o ciclo de vida dos IAM users diretamente na Conta da AWS ou por meio de federação no provedor de identidades de seu quadro de funcionários para o AWS IAM Identity Center. Para IAM users, você pode criar, modificar e excluir usuários e as respectivas permissões associadas na Conta da AWS. No caso de usuários federados, você pode usar o IAM Identity Center para gerenciar o respectivo ciclo de vida sincronizando as informações de usuários e grupos do provedor de identidades da sua organização por meio do protocolo System for Cross-Domain Identity Management (SCIM).

O SCIM é um protocolo de padrão aberto para provisionamento e desprovisionamento automatizados de identidades de usuários em diferentes sistemas. Ao integrar seu provedor de identidades ao IAM Identity Center usando o SCIM, você pode sincronizar automaticamente as informações do usuário e do grupo para ajudar a validar que os privilégios de acesso sejam concedidos, modificados ou revogados com base nas alterações na fonte de identidade autorizada da sua organização.

À medida que as funções e responsabilidades dos funcionários mudam em sua organização, ajuste os respectivos privilégios de acesso de maneira correspondente. Você pode usar os conjuntos de permissões do IAM Identity Center para definir diferentes funções ou responsabilidades de trabalho e associá-las a políticas e permissões apropriadas do IAM. Quando a função de um funcionário muda, você pode atualizar o conjunto de permissões atribuído para refletir as novas responsabilidades. Verifique se ele tem o acesso necessário e segue o princípio de privilégio mínimo.

Etapas da implementação

1. Defina e documente um processo de ciclo de vida do gerenciamento de acesso bem como procedimentos para concessão de acesso inicial, revisões periódicas e desligamento.
2. Implemente perfis, grupos e limites de permissões do IAM para gerenciar o acesso coletivamente e impor os níveis máximos de acesso permitidos.
3. Integre-se a um provedor de identidades federado (como Microsoft Active Directory, Okta, Ping Identity) como fonte confiável para uso de informações de usuários e grupos usando o IAM Identity Center.
4. Use o protocolo SCIM para sincronizar informações de usuários e grupos do provedor de identidades no IAM Identity Center Identity Store.

5. Crie conjuntos de permissões no IAM Identity Center que representem diferentes cargos ou responsabilidades em sua organização. Defina as políticas e permissões apropriadas do IAM para cada conjunto de permissões.
6. Implemente análises regulares de acesso, revogação imediata do acesso e melhoria contínua do processo do ciclo de vida do gerenciamento de acesso.
7. Ofereça treinamento e conhecimento aos funcionários sobre as práticas recomendadas de gerenciamento de acesso.

Recursos

Práticas recomendadas relacionadas:

- [SEC02-BP04 Contar com um provedor de identidades centralizado](#)

Documentos relacionados:

- [Manage your identity source](#)
- [Manage identities in IAM Identity Center](#)
- [Using AWS Identity and Access Management Access Analyzer](#)
- [IAM Access Analyzer policy generation](#)

Vídeos relacionados:

- [AWS re:Inforce 2023 - Manage temporary elevated access with AWS IAM Identity Center](#)
- [AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center](#)
- [AWS re:Invent 2022 - Harness power of IAM policies & rein in permissions w/Access Analyzer](#)

SEC03-BP07 Analisar o acesso público e entre contas

Monitore continuamente as descobertas que destacam o acesso público e entre contas. Reduza o acesso público e o acesso entre contas somente aos recursos específicos que exigem esse acesso.

Resultado desejado: saber quais de seus recursos da AWS são compartilhados e com quem. Monitorar e auditar continuamente seus recursos compartilhados para verificar se eles são compartilhados com apenas entidades principais autorizadas.

Antipadrões comuns:

- Não manter um inventário dos recursos compartilhados.
- Não seguir um processo de aprovação do acesso público ou entre contas aos recursos.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: baixo

Orientação de implementação

Se a sua conta estiver no AWS Organizations, você poderá conceder acesso aos recursos à toda a organização, a unidades organizacionais específicas ou a contas individuais. Se sua conta não for membro de uma organização, você poderá compartilhar recursos com contas individuais. Você pode conceder acesso direto entre contas usando políticas baseadas em recursos, por exemplo, [políticas de buckets do Amazon Simple Storage Service \(Amazon S3\)](#) ou permitindo que uma identidade principal em outra conta assuma um perfil do IAM em sua conta. Ao utilizar políticas de recursos, verifique se o acesso é concedido apenas a entidades principais autorizadas. Defina um processo para aprovar todos os recursos que devem ser acessíveis publicamente.

O [AWS Identity and Access Management Access Analyzer](#) utiliza [segurança demonstrável](#) para identificar todos os caminhos de acesso a um recurso de fora de sua conta. Ele revisa as políticas de recursos continuamente e relata descobertas de acesso público e entre contas para facilitar a análise de acesso potencialmente amplo. Considere configurar o IAM Access Analyzer com o AWS Organizations para verificar se você tem visibilidade a todas as suas contas. O IAM Access Analyzer também possibilita que você [visualize descobertas](#) antes de implantar permissões de recursos. Isso permite validar que as alterações de política concedam apenas o acesso público e entre contas pretendido aos seus recursos. Ao projetar o acesso a várias contas, é possível utilizar [políticas de confiança](#) para controlar em quais casos um perfil pode ser assumido. Por exemplo, você pode usar a chave de condição [PrincipalOrgId para negar uma tentativa de assumir um perfil de fora de seu AWS Organizations](#).

O [AWS Config pode relatar recursos](#) configurados incorretamente, e por meio de verificações de política do AWS Config, pode detectar recursos que tenham acesso público configurado. Serviços, como [AWS Control Tower](#) e [AWS Security Hub](#) simplificam a implantação de controles de detecção e barreiras de proteção nos AWS Organizations para identificar e corrigir recursos publicamente expostos. Por exemplo, o AWS Control Tower tem uma barreira de proteção gerenciada que pode detectar se algum [snapshot do Amazon EBS é restaurado por Contas da AWS](#).

Etapas da implementação

- Pensar em ativar o [AWS Config para AWS Organizations](#): o AWS Config permite que você agregue as descobertas de várias contas em um AWS Organizations em uma conta de administrador delegada. Isso oferece uma visão abrangente e permite que você [implante o Regras do AWS Config nas contas para detectar recursos acessíveis ao público](#).
- Configurar o AWS Identity and Access Management Access Analyzer o IAM Access Analyzer ajuda a identificar os recursos na organização e nas contas, como buckets do Amazon S3 ou perfis do IAM, que são [compartilhados com uma entidade externa](#).
- Usar autocorreção no AWS Config para responder a alterações na configuração do acesso público de buckets do Amazon S3: [é possível reativar automaticamente as configurações de acesso público de bloco para buckets do Amazon S3](#).
- Implementar o monitoramento e os alertas para identificar se os buckets do Amazon S3 se tornaram públicos: é necessário ter o [monitoramento e os alertas](#) implementados para identificar quando o acesso público de blocos do Amazon S3 foi desativado e se os buckets do Amazon S3 se tornaram públicos. Além disso, se você estiver usando o AWS Organizations, poderá criar uma [política de controle de serviços](#) que impeça alterações nas políticas de acesso público do Amazon S3. O AWS Trusted Advisor confere se há buckets do Amazon S3 com permissões de acesso abertas. As permissões de bucket que concedem, upload ou excluem acesso a todos criam possíveis problemas de segurança, pois permitem que qualquer pessoa adicione, modifique ou remova itens em um bucket. A verificação do Trusted Advisor examina as permissões de bucket explícitas e as políticas de bucket associadas que podem substituir as permissões de bucket. Você também pode utilizar o AWS Config para monitorar seus buckets do Amazon S3 para acesso público. Para ter mais informações, consulte [Como usar o AWS Config para monitorar e responder a buckets do Amazon S3 que possibilitam acesso público](#). Ao revisar o acesso, é importante considerar quais tipos de dados estão contidos em buckets do Amazon S3. O [Amazon Macie](#) ajuda a descobrir e proteger dados sigilosos, como PII, PHI e credenciais, como chaves privadas ou da AWS.

Recursos

Documentos relacionados:

- [Usar o AWS Identity and Access Management Access Analyzer](#)
- [Biblioteca de controles do AWS Control Tower](#)
- [Norma de práticas de segurança básicas da AWS](#)
- [Regras gerenciadas do AWS Config](#)

- [Referência de verificação do AWS Trusted Advisor](#)
- [Monitorar resultados da verificação do AWS Trusted Advisor com o Amazon EventBridge](#)
- [Gerenciar regras do AWS Config em todas as contas de sua organização](#)
- [AWS Config e AWS Organizations](#)

Vídeos relacionados:

- [Práticas recomendadas para proteger seu ambiente de várias contas](#)
- [Análise aprofundada do IAM Access Analyzer](#)

SEC03-BP08 Compartilhar recursos com segurança em sua organização

À medida que o número de workloads aumenta, talvez você precise compartilhar o acesso aos recursos nessas workloads ou fornecer os recursos várias vezes nas contas. Você pode ter estruturas para fragmentar seu ambiente, como ter ambientes de desenvolvimento, teste e produção. No entanto, ter estruturas de separação não limita o compartilhamento seguro. Ao compartilhar componentes que se sobrepõem, você pode reduzir a sobrecarga operacional e possibilitar uma experiência consistente sem precisar adivinhar o que ignorou ao criar o mesmo recurso várias vezes.

Resultado desejado: minimizar o acesso acidental utilizando métodos seguros para compartilhar recursos com sua organização e ajudar com sua iniciativa de prevenção de perda de dados. Reduza sua sobrecarga operacional em comparação com o gerenciamento de componentes individuais, reduza os erros gerados pela criação manual do mesmo componente várias vezes e aumente a escalabilidade de suas workloads. É possível se beneficiar da redução de tempo para a resolução em cenários de falhas em vários pontos e aumentar sua confiança na determinação de quando um componente não é mais necessário. Para ter orientações prescritivas sobre como analisar recursos compartilhados externamente, consulte [SEC03-BP07 Analisar o acesso público e entre contas](#).

Antipadrões comuns:

- Falta de um processo para monitorar de forma contínua e alertar automaticamente sobre o compartilhamento externo inesperado.
- Falta de referência sobre o que deve ou não ser compartilhado.
- Ter como padrão uma política amplamente aberta em vez de compartilhar explicitamente quando necessário.
- Criar manualmente recursos básicos que se sobrepõem quando necessário.

Nível de risco exposto se esta prática recomendada não é estabelecida: médio

Orientação de implementação

Projete seus controles e padrões de acesso para reger o consumo de recursos compartilhados com segurança e somente com entidades confiáveis. Monitore recursos compartilhados e revise o acesso a eles de forma contínua e seja alertado sobre o compartilhamento inadequado ou inesperado. Leia [Analisar o acesso público e entre contas](#) para ajudar você a estabelecer a governança a fim de reduzir o acesso externo apenas aos recursos que precisam dele e estabelecer um processo para monitorar de forma contínua e alertar automaticamente.

O compartilhamento entre contas no AWS Organizations é compatível com [uma série de serviços da AWS](#), como o [AWS Security Hub](#), [Amazon GuardDuty](#) e o [AWS Backup](#). Esses serviços possibilitam compartilhar os dados em uma conta central, acessá-los ou gerenciar recursos e dados dessa conta. Por exemplo, o AWS Security Hub pode transferir as descobertas de contas individuais para uma conta central onde é possível visualizar todas elas. O AWS Backup pode realizar um backup de um recurso e compartilhá-lo entre contas. É possível utilizar o [AWS Resource Access Manager](#) (AWS RAM) para compartilhar outros recursos comuns, como [sub-redes de VPC e anexos do Transit Gateway](#), [AWS Network Firewall](#) ou pipelines [Amazon SageMaker](#).

Para restringir sua conta para somente compartilhar recursos em sua organização, utilize [políticas de controle de serviços \(SCPs\)](#) para impedir o acesso a entidades principais externas. Ao compartilhar recursos, combine controles baseados em identidade e controles de rede para [criar um perímetro de dados para sua organização](#) a fim de ajudar a proteger contra o acesso acidental. Um perímetro de dados é um conjunto de barreiras de proteção preventivas que ajudam a garantir que apenas suas identidades confiáveis acessem recursos confiáveis das redes esperadas. Esses controles impõem limites apropriados sobre quais recursos podem ser compartilhados e impedir o compartilhamento ou a exposição de recursos que não devem ser permitidos. Por exemplo, como parte de um perímetro de dados, é possível usar políticas de endpoint de VPC e a condição `AWS:PrincipalOrgId` para garantir que as identidades que acessam seus buckets do Amazon S3 pertençam à sua organização. É importante observar que as [SCPs não se aplicam a perfis vinculados a serviço \(LSR\) nem a entidades principais de serviços da AWS](#).

Ao utilizar o Amazon S3, [desative as ACLs de seu bucket do Amazon S3](#) e utilize políticas do IAM para definir o controle de acesso. Para [restringir o acesso a uma origem do Amazon S3](#) a partir do [Amazon CloudFront](#), migre da identidade do acesso de origem (OAI) para um controle de acesso de origem (OAC), que é compatível com recursos adicionais, por exemplo, a criptografia do lado do servidor com o [AWS Key Management Service](#).

Em alguns casos, convém permitir o compartilhamento de recursos fora de sua organização ou conceder a terceiros acesso aos seus recursos. Para ter orientações prescritivas sobre o gerenciamento de permissões para compartilhar recursos externamente, consulte [Gerenciamento de permissões](#).

Etapas da implementação

1. Utilize o AWS Organizations.

O AWS Organizations é um serviço de gerenciamento de contas que permite consolidar várias Contas da AWS em uma organização que você cria e gerencia centralmente. É possível agrupar suas contas em unidades organizacionais (UOs) e anexar políticas diferentes a cada UO a fim de ajudar a atender às suas necessidades orçamentárias, de segurança e conformidade. Também é possível controlar como serviços de inteligência artificial (IA) e machine learning (ML) da AWS podem coletar e armazenar dados e usar o gerenciamento de várias contas dos serviços da AWS integrados ao Organizations.

2. Integre o AWS Organizations aos serviços da AWS.

Ao ativar um serviço da AWS para realizar tarefas em seu nome nas contas membros de sua organização, o AWS Organizations cria um perfil vinculado a serviço do IAM para esse serviço em cada conta membro. Você deve gerenciar o acesso confiável usando o AWS Management Console, as APIs da AWS ou a AWS CLI. Para ter orientações prescritivas sobre como ativar o acesso confiável, consulte [Usar o AWS Organizations com outros serviços da AWS](#) e [Serviços da AWS que podem ser usados com o Organizations](#).

3. Estabeleça um perímetro de dados.

O perímetro da AWS, geralmente, é representado como uma organização gerenciada pelo AWS Organizations. Junto com redes e sistemas on-premises, o acesso a recursos da AWS é o que muitos consideram o perímetro de My AWS. O objetivo do perímetro é garantir que o acesso seja permitido se a identidade e o recurso forem confiáveis e a rede for esperada.

a. Defina e implante os perímetros.

Siga as etapas descritas em [Implementação do perímetro](#) do whitepaper Criar um perímetro na AWS para cada condição de autorização. Para ter orientações prescritivas sobre como proteger a camada de rede, consulte [Proteção de redes](#).

b. Monitore e alerte de forma contínua.

- O [AWS Identity and Access Management Access Analyzer](#) ajuda a identificar os recursos na organização e nas contas que são compartilhados com entidades externas. É possível integrar o [IAM Access Analyzer ao AWS Security Hub](#) para enviar e agregar as descobertas para um recurso do IAM Access Analyzer para o Security Hub a fim de ajudar a analisar o procedimento de segurança de seu ambiente. Para ativar a integração, ative o IAM Access Analyzer e o Security Hub em cada região em cada conta. Também é possível utilizar o Regras do AWS Config para fazer auditoria da configuração e alertar a parte adequada utilizando o [AWS Chatbot com o AWS Security Hub](#). Depois, você pode utilizar [Documentos de automação do AWS Systems Manager](#) para corrigir os recursos sem conformidade.
- c. Para ter orientações prescritivas sobre como monitorar e alertar de forma contínua sobre recursos compartilhados externamente, consulte [Analisar o acesso público e entre contas](#).
4. Utilize o compartilhamento de recursos em serviços da AWS e restrinja-o adequadamente.

Muitos serviços da AWS possibilitam compartilhar recursos com outra conta ou almejar um recurso em outra conta, como [Imagens de máquina da Amazon \(AMIs\)](#) e [AWS Resource Access Manager \(AWS RAM\)](#). Restrinja a API `ModifyImageAttribute` para especificar as contas confiáveis com as quais compartilhar a AMI. Especifique a condição `ram:RequestedAllowsExternalPrincipals` ao utilizar o AWS RAM para restringir o compartilhamento somente à sua organização, a fim de ajudar a impedir o acesso de identidades não confiáveis. Para ter orientações prescritivas e considerações, consulte [Compartilhamento de recursos e destinos externos](#).

5. Utilize o AWS RAM para compartilhar com segurança em uma conta ou com outras Contas da AWS.

O [AWS RAM](#) ajuda você a compartilhar com segurança os recursos criados com perfis e usuários em sua conta e com outras Contas da AWS. Em um ambiente de várias contas, o AWS RAM possibilita criar um recurso uma vez e compartilhá-lo com outras contas. Essa abordagem ajuda a reduzir sua sobrecarga operacional ao oferecer consistência, visibilidade e capacidade de auditoria por meio de integrações com o Amazon CloudWatch e o AWS CloudTrail, o que você não recebe ao utilizar o acesso entre contas.

Se você tiver recursos compartilhados anteriormente com o uso de uma política baseada em recurso, é possível utilizar a API [PromoteResourceShareCreatedFromPolicy](#) ou equivalente a fim de promover o compartilhamento de recursos para um compartilhamento completo de recursos do AWS RAM.

Em alguns casos, convém realizar etapas adicionais para compartilhar recursos. Por exemplo, para compartilhar um snapshot criptografado, é necessário [compartilhar uma chave do AWS KMS](#).

Recursos

Práticas recomendadas relacionadas:

- [SEC03-BP07 Analisar o acesso público e entre contas](#)
- [SEC03-BP09 Compartilhar recursos com segurança com terceiros](#)
- [SEC05-BP01 Criar camadas de rede](#)

Documentos relacionados:

- [Proprietário do bucket concede permissão entre contas a objetos que não possui](#)
- [Como usar políticas de confiança com o IAM](#)
- [Criar um perímetro de dados na AWS](#)
- [Como usar um ID externo ao conceder acesso aos seus recursos da AWS para terceiros](#)
- [Serviços da AWS que podem ser usados com o AWS Organizations](#)
- [Estabelecer um perímetro de dados na AWS: permitir apenas que identidades confiáveis acessem os dados da empresa](#)

Vídeos relacionados:

- [Acesso granular com o AWS Resource Access Manager](#)
- [Como proteger seu perímetro de dados com endpoints da VPC](#)
- [Estabelecer um perímetro de dados na AWS](#)

Ferramentas relacionadas:

- [Exemplos de política de perímetro de dados](#)

SEC03-BP09 Compartilhar recursos com segurança com terceiros

A segurança de seu ambiente de nuvem não é interrompida em sua organização. Sua organização pode contar com terceiros para gerenciar uma parte de seus dados. O gerenciamento de permissões para o sistema gerenciado por terceiros deve seguir a prática de acesso just-in-time utilizando o princípio de privilégio mínimo com credenciais temporárias. Ao trabalhar em parceria com terceiros, é possível reduzir o escopo do impacto e o risco de acesso acidental.

Resultado desejado: credenciais do AWS Identity and Access Management (IAM) de longo prazo, chaves de acesso do IAM e chaves secretas associadas a um usuário podem ser usadas por qualquer pessoa desde que as credenciais sejam válidas e ativas. O uso de um perfil do IAM e credenciais temporárias ajuda você a melhorar seu procedimento de segurança geral reduzindo o esforço para manter credenciais de longo prazo, inclusive o gerenciamento e a sobrecarga operacional dessas informações sigilosas. Ao utilizar um identificador universalmente exclusivo (UUID) para o ID externo na política de confiança do IAM e manter as políticas do IAM anexadas ao perfil do IAM sob seu controle, é possível fazer auditoria e garantir que o acesso concedido a terceiros não seja permissivo demais. Para ter orientações prescritivas sobre como analisar recursos compartilhados externamente, consulte [SEC03-BP07 Analisar o acesso público e entre contas](#).

Antipadrões comuns:

- Utilizar a política de confiança do IAM padrão sem condições.
- Utilizar credenciais e chaves de acesso de longo prazo do IAM.
- Reutilizar IDs externos.

Nível de risco exposto se esta prática recomendada não é estabelecida: médio

Orientação de implementação

Talvez você deseje permitir o compartilhamento de recursos fora do AWS Organizations ou conceder a terceiros acesso à sua conta. Por exemplo, um parceiro (terceiros) pode oferecer uma solução de monitoramento que precise acessar recursos em sua conta. Nesses casos, crie um perfil entre contas do IAM somente com os privilégios necessários para o parceiro. Além disso, defina uma política de segurança com o uso da [condição de ID externo](#). Ao utilizar um ID externo, você ou o parceiro pode gerar um ID exclusivo para cada cliente, terceiros ou locação. O ID exclusivo não deve ser controlado por ninguém, exceto por você, depois de criado. O parceiro deve implementar um processo para relacionar o ID externo ao cliente de forma segura, auditável e reproduzível.

Também é possível usar o [IAM Roles Anywhere](#) para gerenciar perfis do IAM para aplicações fora do AWS que utilizam APIs da AWS.

Se o parceiro não precisar mais de acesso ao seu ambiente, remova o perfil. Evite fornecer credenciais de longo prazo para terceiros. Esteja ciente de outros serviços da AWS compatíveis com o compartilhamento. Por exemplo, o AWS Well-Architected Tool possibilita o [compartilhamento de uma workload](#) com outras Contas da AWS, e o [AWS Resource Access Manager](#) ajuda você a compartilhar com segurança um recurso da AWS que você possua com outras contas.

Etapas da implementação

1. Utilize perfis entre contas para fornecer acesso a contas externas.

Os [perfis entre contas](#) reduzem a quantidade de informações sigilosas armazenadas por contas externas e terceiros para atender aos clientes. Os perfis entre contas possibilitam a você conceder acesso a recursos da AWS em sua conta de forma segura a terceiros, como AWS Partners ou outras contas em sua organização e, ao mesmo tempo, manter a capacidade de gerenciar e auditar esse acesso.

O parceiro pode oferecer serviço a você a partir de uma infraestrutura híbrida ou, como alternativa, extrair dados de um local externo. O [IAM Roles Anywhere](#) ajuda você a possibilitar que workloads de terceiros interajam com segurança com suas workloads da AWS e reduzir ainda mais a necessidade de credenciais de longo prazo.

Você não deve usar credenciais ou chaves de acesso de longo prazo associadas a usuários para conceder acesso a contas externas. Em vez disso, utilize perfis entre contas para conceder acesso entre contas.

2. Utilize um ID externo com terceiros.

O uso de um [ID externo](#) possibilita designar quem pode assumir um perfil em uma política de confiança do IAM. A política de confiança pode exigir que o usuário que assume o perfil imponha a condição e o destino no qual ele está operando. Ele também fornece uma maneira para que o proprietário da conta permita que a função seja assumida somente em circunstâncias específicas. A função principal do ID externo é resolver e evitar o problema de [substituto confuso](#).

Utilize um ID externo se você for proprietário de uma Conta da AWS e tiver configurado um perfil para terceiros que acesse outras Contas da AWS além da sua, ou quando você pode assumir perfis em nome de clientes diferentes. Trabalhe com terceiros ou a AWS Partner para estabelecer uma condição de ID externo a ser incluída na política de confiança do IAM.

3. Utilize IDs externos universalmente exclusivos.

Implemente um processo que gere um valor exclusivo aleatório para um ID externo, como um identificador universalmente exclusivo (UUID). Um parceiro que reutilize IDs externos entre diferentes clientes não resolve o problema de substituto confuso porque o cliente A pode ser capaz de visualizar dados do cliente B utilizando o ARN do perfil do cliente B junto com o ID externo duplicado. Em um ambiente de vários locatários, em que um parceiro atende a vários clientes com diferentes Contas da AWS, o parceiro deve usar um ID exclusivo diferente como o ID externo de cada Conta da AWS. O parceiro é responsável por detectar IDs externos duplicados e mapear de forma segura cada cliente ao seu respectivo ID externo. O parceiro deve testar para verificar se ele pode assumir o perfil somente ao especificar o ID externo. O parceiro deve evitar armazenar o ARN do perfil do cliente e o ID externo até que este seja necessário.

O ID externo não é tratado como segredo, mas ele não pode ser um valor facilmente dedutível, como um número de telefone, um nome ou o ID da conta. Torne o ID externo um campo somente leitura de forma que o ID externo não possa ser alterado com o fim de representar a configuração.

Você ou o parceiro podem gerar o ID externo. Defina um processo para determinar quem é responsável pela geração do ID. Seja qual for a entidade que crie o ID externo, o parceiro impõe a exclusividade e os formatos de forma consistente entre os clientes.

4. Deprecie credenciais de longo prazo fornecidas pelo cliente.

Deprecie o uso de credenciais de longo prazo e use perfis entre clientes ou o IAM Roles Anywhere. Se você precisar utilizar credenciais de longo prazo, estabeleça um plano para migrar para um acesso baseado em perfil. Para obter detalhes sobre como gerenciar chaves, consulte [Gerenciamento de identidades](#). Trabalhe também com a equipe de sua Conta da AWS e o parceiro para estabelecer um runbook de mitigação de riscos. Para ter orientações prescritivas sobre como responder e mitigar o impacto em potencial do incidente de segurança, consulte [Resposta a incidentes](#).

5. Verifique se a configuração tem orientações prescritivas ou é automatizada.

A política criada para acesso entre contas em suas contas deve seguir o [princípio de privilégio mínimo](#). O parceiro deve fornecer um documento de política de perfil ou um mecanismo de configuração automatizada que utilize um modelo do AWS CloudFormation ou um equivalente para você. Isso reduz a chance de erros associados à criação manual de políticas e oferece uma trilha auditável. Para ter mais informações sobre como usar um modelo do AWS CloudFormation para criar perfis entre contas, consulte [Perfis entre contas](#).

O parceiro deve fornecer um mecanismo de configuração automatizado e auditável. No entanto, ao utilizar o documento de política de perfis que descreve o acesso necessário, você deve automatizar a configuração do perfil. Com um modelo do AWS CloudFormation ou equivalente, você deve monitorar alterações com detecção de desvios como parte da prática de auditoria.

6. Considere alterações.

Sua estrutura de contas, sua necessidade de terceiros ou a oferta de serviço pode ser alterada. Você deve antecipar alterações e falhas e planejar adequadamente com as pessoas, o processo e a tecnologia corretos. Audite o nível de acesso que você concede periodicamente e implemente métodos de detecção para alertar você de alterações inesperadas. Monitore e audite o uso do perfil e o datastore dos IDs externos. Você deve estar preparado para revogar o acesso de terceiros, seja de forma temporária ou permanente, como resultado de alterações ou padrões de acesso inesperados. Além disso, meça o impacto de sua operação de revogação, inclusive o tempo para realizá-la, as pessoas envolvidas, o custo e o impacto de outros recursos.

Para ter orientações prescritivas sobre métodos de detecção, consulte as [Práticas recomendadas de detecção](#).

Recursos

Práticas recomendadas relacionadas:

- [SEC02-BP02 Usar credenciais temporárias](#)
- [SEC03-BP05 Definir barreiras de proteção de permissões para sua organização](#)
- [SEC03-BP06 Gerenciar o acesso com base no ciclo de vida](#)
- [SEC03-BP07 Analisar o acesso público e entre contas](#)
- [SEC04 Detecção](#)

Documentos relacionados:

- [Proprietário do bucket concede permissão entre contas a objetos que não possui](#)
- [Como usar políticas de confiança com os perfis do IAM](#)
- [Delegar acesso entre Contas da AWS usando funções do IAM](#)
- [Como acesso recursos em outra Conta da AWS usando o IAM?](#)
- [Práticas recomendadas de segurança no IAM](#)

- [Lógica de avaliação de política entre contas](#)
- [Como usar um ID externo ao conceder acesso a seus recursos da AWS a terceiros](#)
- [Coletar informações de recursos do AWS CloudFormation criados em contas externas com recursos personalizados](#)
- [Usar ID externo com segurança para acessar contas da AWS pertencentes a outros](#)
- [Estender perfis do IAM fora do IAM com IAM Roles Anywhere\)](#)

Vídeos relacionados:

- [Como permito que usuários ou perfis em uma Conta da AWS separada acessem minha Conta da AWS?](#)
- [AWS re:Invent 2018: Torne-se um mestre em políticas do IAM em 60 minutos ou menos](#)
- [AWSPráticas recomendadas do IAM e decisões de design](#)

Exemplos relacionados:

- [Well-Architected Lab: Assumir perfil do IAM entre contas do Lambda \(Nível 300\)](#)
- [Configurar o acesso entre contas ao Amazon DynamoDB](#)
- [AWS STS Network Query Tool](#)

Detecção

A detecção consiste em duas partes: a detecção de alterações de configuração inesperadas ou indesejadas e a detecção de comportamento inesperado. A primeira pode ocorrer em vários locais em um ciclo de vida de entrega de aplicações. Usando a infraestrutura como código (por exemplo, um modelo do CloudFormation), é possível verificar a configuração indesejada antes que uma workload seja implantada ao aplicar as verificações nos pipelines de CI/CD ou controle de origem. Em seguida, ao implantar uma workload em ambientes de não produção e de produção, você pode verificar a configuração usando ferramentas nativas da AWS, de código aberto ou de parceiros da AWS. Essas verificações podem ser para configuração que não atende aos princípios de segurança ou práticas recomendadas, ou para alterações que foram feitas entre uma configuração testada e implementada. Para uma aplicação em execução, é possível verificar se a configuração foi alterada de maneira inesperada, inclusive fora de uma implantação conhecida ou de um evento de escalabilidade automatizada.

Para a segunda parte da detecção, comportamento inesperado, é possível usar ferramentas ou alertar sobre um aumento em um tipo específico de chamada de API. Usando o Amazon GuardDuty, você pode receber um alerta quando ocorrer atividade inesperada e potencialmente não autorizada ou maliciosa em suas contas da AWS. Monitore também explicitamente as chamadas de API mutantes que você não espera que sejam usadas na workload e as chamadas de API que alteram a postura de segurança.

Detecção permite identificar uma possível configuração incorreta de segurança, uma ameaça ou um comportamento inesperado. Essa é uma parte essencial do ciclo de vida de segurança e pode ser usada para apoiar um processo de qualidade, uma obrigação legal ou de conformidade e os esforços de identificação e resposta a ameaças. Existem diferentes tipos de mecanismos de detecção. Por exemplo, os logs da carga de trabalho podem ser analisados em busca de explorações que estão sendo usadas. Você deve revisar regularmente os mecanismos de detecção relacionados à sua carga de trabalho para garantir que esteja atendendo às políticas e aos requisitos internos e externos. Os alertas e notificações automatizados devem se basear em condições definidas para permitir que as equipes ou ferramentas investiguem. Esses mecanismos são fatores reativos importantes que podem ajudar a organização a identificar e entender o escopo de atividades anômalas.

Na AWS, há várias abordagens para lidar com mecanismos de detecção. As seções a seguir descrevem como usar essas abordagens:

Práticas recomendadas

- [SEC04-BP01 Configurar registro em log de serviço e aplicação](#)
- [SEC04-BP02 Capturar logs, descobertas e métricas em locais padronizados](#)
- [SEC04-BP03 Correlacionar e enriquecer os alertas de segurança](#)
- [SEC04-BP04 Iniciar a correção de recursos irregulares](#)

SEC04-BP01 Configurar registro em log de serviço e aplicação

Retenha logs de eventos de segurança de serviços e aplicações. Esse é um princípio fundamental de segurança para auditoria, investigações e casos de uso operacionais e um requisito de segurança comum orientado por padrões, políticas e procedimentos de governança, risco e conformidade (GRC).

Resultado desejado: uma organização deve ser capaz de recuperar de forma confiável e consistente logs de eventos de segurança de serviços e aplicações da AWS de modo pontual quando necessário a fim de cumprir um processo ou obrigação interna, como resposta a incidentes de segurança. Considere centralizar os logs para ter melhores resultados operacionais.

Antipadrões comuns:

- Os logs são armazenados de forma perpétua ou excluídos muito precocemente.
- Todos podem acessar os logs.
- Contar inteiramente com processos manuais para uso e governança de logs.
- Armazenar todos os tipos de log em caso de necessidade.
- Conferir a integridade dos logs apenas quando necessário.

Benefícios do estabelecimento desta prática recomendada: implementar um mecanismo de análise da causa raiz (RCA) para incidentes de segurança e uma fonte de evidências para suas obrigações de governança, risco e conformidade.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientação de implementação

Durante uma investigação de segurança ou outros casos de uso com base em seus requisitos, você precisa ser capaz de analisar os logs relevantes a fim de registrar e entender o escopo total e a linha do tempo do incidente. Os logs também são necessários para geração de alertas indicando que

ocorrerem determinadas ações de interesse. É essencial selecionar, ativar, armazenar e configurar mecanismos de consulta, recuperação e alertas.

Etapas da implementação

- Selecione e ative fontes de logs. Antes de uma investigação de segurança, você precisa capturar logs relevantes para reconstruir, de forma retroativa a atividade em uma Conta da AWS. Selecione e ative fontes de logs relevantes para suas workloads.

Os critérios de seleção de fonte de logs devem se basear nos casos de uso necessários à sua empresa. Estabeleça uma trilha para cada Conta da AWS utilizando o AWS CloudTrail ou uma trilha de AWS Organizations e configure um bucket do Amazon S3 para ela.

O AWS CloudTrail é um serviço de registro em log que rastreia chamadas de API feitas em uma Conta da AWS capturando a atividade do serviço da AWS. É ativado por padrão com uma retenção de 90 dias de eventos de gerenciamento que podem ser [recuperados por meio do histórico de eventos do CloudTrail](#) utilizando o AWS Management Console, a AWS CLI ou um AWS SDK. Para ter uma retenção maior e visibilidade dos eventos de dados, [crie uma trilha do CloudTrail](#) e associe-a a um bucket do Amazon S3 e opcionalmente com um grupo de logs do Amazon CloudWatch. Como alternativa, você pode criar um [CloudTrail Lake](#), que retém logs do CloudTrail por até sete anos e oferece um recursos e consultas baseado em SQL

A AWS recomenda que os clientes que utilizam uma VPC ativem o tráfego de rede e os logs de DNS por meio dos [Logs de fluxo de VPC](#) e dos [logs de consultas do Amazon Route 53 Resolver](#), respectivamente, transmitindo-os a um bucket do Amazon S3 ou a um grupo de logs do CloudWatch. É possível criar um log de fluxo de VPC, uma sub-rede ou uma interface de rede. Para logs de fluxo de VPC, é possível ser seletivo em relação a como e onde usar os logs de fluxo para reduzir o custo.

Logs do AWS CloudTrail, Logs de fluxo de VPC e logs de consulta do Route 53 Resolver são as fontes básicas de registro em log para oferecer compatibilidade com investigações de segurança na AWS. Também é possível usar o [Amazon Security Lake](#) para coletar, normalizar e armazenar esses dados de logs no formato do Apache Parquet e no Open Cybersecurity Schema Framework (OCSF), que estão prontos para consulta. O Security Lake também é compatível com outros logs da AWS e logs de fontes de terceiros.

Os serviços da AWS podem gerar logs não capturados pelas fontes de log básicas, como logs do Elastic Load Balancing, logs do AWS WAF, logs de gravador do AWS Config, descobertas do Amazon GuardDuty, logs de auditoria do Amazon Elastic Kubernetes Service (Amazon EKS)

e logs de aplicações e do sistema de instâncias do Amazon EC2. Para ter uma lista completa de opções de registro em log e monitoramento, consulte [Apêndice A: Definições de recursos de nuvem: registro em log e eventos](#) do [Guia de resposta a incidentes de segurança da AWS](#).

- Recursos de registro em log de pesquisa para cada serviço e aplicação da AWS: cada serviço e aplicação da AWS oferecem opções armazenamento de logs, sendo cada um com seus próprios recursos de retenção e ciclo de vida. Os dois serviços de armazenamento de logs mais comuns são Amazon Simple Storage Service (Amazon S3) e Amazon CloudWatch. Para períodos de retenção longos, é recomendável utilizar o Amazon S3 para seus recursos de economia e ciclo de vida flexíveis. Se a opção de registro em log principal for logs do Amazon CloudWatch, como opção, você deve considerar o arquivamento de logs menos acessados no Amazon S3.
- Selecione o armazenamento de logs: a escolha do armazenamento de logs, geralmente, é relacionada a qual ferramenta de consultas você utiliza, recursos de retenção, familiaridade e custo. As principais opções para armazenamento de logs são um bucket do Amazon S3 ou um grupo de logs do CloudWatch.

Um bucket do Amazon S3 oferece armazenamento econômico e durável com uma política de ciclo de vida opcional. Os logs armazenados em buckets do Amazon S3 podem ser consultados com serviços como o Amazon Athena.

Um grupo de logs do CloudWatch oferece armazenamento durável e um recurso de consultas incorporado por meio do CloudWatch Logs Insights.

- Identifique a retenção de logs apropriada: quando você utiliza um bucket do Amazon S3 ou o grupo de logs do CloudWatch para armazenar logs, é necessário estabelecer ciclos de vida adequados para cada fonte de logs a fim de otimizar os custos de armazenamento e recuperação. Os clientes geralmente têm entre três meses a um ano de logs prontamente disponíveis para consultas, com retenção de até sete anos. A escolha de disponibilidade e retenção deve se alinhar aos seus requisitos de segurança e um composto de atribuições regulatórias, estatutárias e de negócios.
- Ative o registro em log para cada serviço e aplicação da AWS com políticas adequadas de retenção e ciclo de vida: para cada serviço ou aplicação da AWS em sua organização, procure as orientações específicas de configuração de registro em log:
 - [Configurar a trilha do AWS CloudTrail](#)
 - [Configurar logs de fluxo de VPC](#)
 - [Configurar as exportações de descobertas do Amazon GuardDuty](#)
 - [Configurar os registros do AWS Config](#)

- [Configurar o tráfego de ACL da web do AWS WAF](#)
 - [Configurar os logs de tráfego de rede do AWS Network Firewall](#)
 - [Configurar logs de acesso do Elastic Load Balancing](#)
 - [Configurar logs de consulta do Amazon Route 53 resolver](#)
 - [Configurar logs do Amazon RDS](#)
 - [Configurar logs do ambiente de gerenciamento Amazon EKS](#)
 - [Configurar o agente do Amazon CloudWatch para instâncias do Amazon EC2 e servidores on-premises](#)
- Selecione e implemente os mecanismos de consulta para logs: para consultas de log, você pode usar o [CloudWatch Logs Insights](#) para dados armazenados em grupos de logs do CloudWatch, e o [Amazon Athena](#) e o [Amazon OpenSearch Service](#) para dados armazenados no Amazon S3. Também é possível usar ferramentas de consulta de terceiros, como um serviço de gerenciamento de eventos e informações de segurança (SIEM).

O processo para selecionar uma ferramenta de consulta de log deve considerar as pessoas, o processo e os aspectos de tecnologia de suas operações de segurança. Selecione uma ferramenta que atenda aos requisitos operacionais, de negócios e segurança, esteja acessível e possa receber manutenção no longo prazo. Lembre-se de que as ferramentas de consulta de logs funcionam da forma ideal quando o número de logs a serem verificados é mantido dentro dos limites da ferramenta. Não é incomum ter várias ferramentas de consulta devido a restrições financeiras ou técnicas.

Por exemplo, você pode usar uma ferramenta de gerenciamento de eventos e informações de segurança (SIEM) de terceiros para realizar consultas para os últimos 90 dias de dados, mas usar o Athena para realizar consultas além de 90 dias devido ao custo de ingestão de logs de um SIEM. Seja qual for a implementação, garanta que sua abordagem minimize o número de ferramentas necessárias para maximizar a eficiência operacional, especialmente durante a investigação de um evento de segurança.

- Use logs para alertas: a AWS oferece alertas por meio de vários serviços de segurança:
 - O [AWS Config](#) monitora e registra as configurações de recursos da AWS e permite automatizar as tarefas de avaliação e correção em relação às configurações desejadas.
 - O [Amazon GuardDuty](#) é um serviço de detecção de ameaças que monitora de forma contínua a existência de atividade mal-intencionada e comportamento não autorizado para proteger suas Contas da AWS e workloads. O GuardDuty ingere, agrega e analisa informações de fontes, como eventos de dados e gerenciamento do AWS CloudTrail, logs de DNS, logs de

fluxo de VPC e logs do Amazon EKS Audit. O GuardDuty extrai fluxos de dados independentes diretamente do CloudTrail, de logs de fluxo de VPC, logs de consulta ao DNS e do Amazon EKS. Não é necessário gerenciar políticas de bucket do Amazon S3 nem modificar a forma de coletar e armazenar logs. Ainda é recomendável reter esses logs para sua própria investigação e fins de conformidade.

- O [AWS Security Hub](#) fornece um único local que agrega, organiza e prioriza alertas de segurança ou descobertas de vários serviços da AWS e produtos opcionais de terceiros para oferecer uma visão abrangente dos alertas de segurança e do status de conformidade.

Você também pode utilizar mecanismos de geração de alertas personalizados para alertas de segurança não cobertos por esses serviços ou para alertas específicos relevantes para o seu ambiente. Para ter informações sobre a criação desses alertas e detecções, consulte [Detecção no Guia de resposta a incidentes de segurança da AWS](#).

Recursos

Práticas recomendadas relacionadas:

- [SEC04-BP02 Capturar logs, descobertas e métricas em locais padronizados](#)
- [SEC07-BP04 Definir o gerenciamento escalável do ciclo de vida dos dados](#)
- [SEC10-BP06 Pré-implantação de ferramentas](#)

Documentos relacionados:

- [Guia de resposta a incidentes de segurança da AWS](#)
- [Conceitos básicos do Amazon Security Lake](#)
- [Conceitos básicos: Amazon CloudWatch Logs](#)
- [Soluções de segurança parceiros: registro em log e monitoramento](#)

Vídeos relacionados:

- [AWS re:Invent 2022: Introdução ao Amazon Security Lake](#)

Exemplos relacionados:

- [Assisted Log Enabler for AWS](#)

- [Exportação histórica de descobertas do AWS Security Hub](#)

Ferramentas relacionadas:

- [Snowflake for Cybersecurity](#)

SEC04-BP02 Capturar logs, descobertas e métricas em locais padronizados

As equipes de segurança confiam em logs e descobertas para analisar eventos que podem indicar atividades não autorizadas ou alterações não intencionais. Para agilizar essa análise, capture logs e descobertas de segurança em locais padronizados. Isso disponibiliza pontos de dados de interesse para correlação e pode simplificar a integração de ferramentas.

Resultado desejado: você tem uma abordagem padronizada para coletar, analisar e visualizar dados de logs, descobertas e métricas. As equipes de segurança podem correlacionar, analisar e visualizar com eficiência os dados de segurança em sistemas diferentes para descobrir possíveis eventos de segurança e identificar anomalias. Os sistemas de gerenciamento de eventos e informações de segurança (SIEM) ou outros mecanismos são integrados para consultar e analisar dados de log e oferecer respostas oportunas, rastreamento e encaminhamento de eventos de segurança.

Antipadrões comuns:

- As equipes são proprietárias e gerenciam de forma independente o registro em log e a coleta de métricas que são inconsistentes com a estratégia de registro em log da organização.
- As equipes não têm controles de acesso adequados para restringir a visibilidade e a alteração dos dados coletados.
- As equipes não controlam logs, descobertas e métricas de segurança como parte da política de classificação de dados.
- As equipes negligenciam os requisitos de soberania e localização dos dados ao configurar as coletas de dados.

Benefícios do estabelecimento desta prática recomendada: uma solução de registro em log padronizada para coletar e consultar dados de log e eventos melhora os insights deduzidos das informações que eles contêm. Configurar um ciclo de vida automatizado para os dados de log coletados pode reduzir os custos incorridos pelo armazenamento de logs. Você pode criar um

controle de acesso refinado para as informações de log coletadas de acordo com a confidencialidade dos dados e os padrões de acesso necessários para as equipes. Você pode integrar ferramentas para correlacionar, visualizar e obter insights dos dados.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

O aumento do uso da AWS em uma organização resulta em um número crescente de workloads e ambientes distribuídos. Como cada um desses ambientes e workloads gera dados sobre as atividades dentro deles, capturar e armazenar esses dados localmente representa um desafio para as operações de segurança. As equipes de segurança usam ferramentas, como sistemas de gerenciamento de eventos e informações de segurança (SIEM), para coletar dados de fontes distribuídas e submetê-los a fluxos de trabalho de correlação, análise e resposta. Isso requer o gerenciamento de um conjunto complexo de permissões para acessar as várias fontes de dados e despesas operacionais indiretas adicionais na operação dos processos de extração, transformação e carregamento (ETL).

Para superar esses desafios, considere agregar todas as fontes relevantes de dados de log de segurança em uma conta de [arquivo de logs](#), conforme descrito em [Organizing Your AWS Environment Using Multiple Accounts](#). Isso inclui todos os dados relacionados à segurança da workload e os logs gerados por serviços da AWS, como [AWS CloudTrail](#), [AWS WAF](#), [Elastic Load Balancing](#) e [Amazon Route 53](#). Há vários benefícios em capturar esses dados em locais padronizados em uma Conta da AWS separada com as devidas permissões entre contas. Essa prática ajuda a evitar a violação de logs em workloads e ambientes comprometidos, fornece um único ponto de integração para ferramentas adicionais e oferece um modelo mais simplificado para configurar a retenção de dados e o ciclo de vida. Avalie os impactos da soberania de dados, dos escopos de conformidade e de outras regulamentações para determinar se vários locais de armazenamento de dados de segurança e períodos de retenção são necessários.

Para facilitar a captura e padronização de logs e descobertas, avalie o [Amazon Security Lake](#) em sua conta de arquivo de logs. Você pode configurar o Security Lake para ingerir automaticamente os dados de fontes comuns, como o CloudTrail, o Route 53, o [Amazon EKS](#) e [logs de fluxo da VPC](#). Você também pode configurar o AWS Security Hub como fonte de dados no Security Lake, o que permite que você correlacione descobertas de outros serviços da AWS, como o [Amazon GuardDuty](#) e o [Amazon Inspector](#), com seus dados de log. Você também pode usar integrações de fontes de dados de terceiros ou configurar fontes de dados personalizadas. Todas as integrações padronizam os dados no formato [Open Cybersecurity Schema Framework](#) (OCSF) e são armazenados em

[buckets do Amazon S3](#) como arquivos Parquet, eliminando a necessidade de processamento de ETL.

Com o armazenamento de dados de segurança em locais padronizados, é possível contar com recursos avançados de análise. A AWS recomenda que você implante as ferramentas para análise de segurança que operam em um ambiente da AWS em uma conta de [ferramentas de segurança](#) separada da conta de arquivo de logs. Essa abordagem permite implantar controles detalhados para proteger a integridade e a disponibilidade dos logs e do processo de gerenciamento de logs, diferentemente das ferramentas que os acessam. Considere usar serviços, como o [Amazon Athena](#), para executar consultas sob demanda que correlacionam várias fontes de dados. Você também pode integrar ferramentas de visualização, como o [Amazon QuickSight](#). As soluções baseadas em IA estão se tornando cada vez mais disponíveis e podem desempenhar funções como traduzir descobertas em resumos legíveis por humanos e interação em linguagem natural. Essas soluções geralmente são mais fáceis de integrar por terem um local de armazenamento de dados padronizado para consulta.

Etapas da implementação

1. Crie as contas de arquivo de logs e de ferramentas de segurança
 - a. Usando o AWS Organizations, [crie as contas de arquivo de logs e de ferramentas de seguranças](#) em uma unidade organizacional de segurança. Se estiver usando o AWS Control Tower para gerenciar sua organização, as contas de arquivo de logs e de ferramentas de segurança serão criadas automaticamente para você. Configure funções e permissões para acessar e administrar essas contas conforme necessário.
2. Configure locais de dados de segurança padronizados
 - a. Determine sua estratégia para criar locais de dados de segurança padronizados. Você pode conseguir isso por meio de opções como abordagens comuns de arquitetura de data lake, produtos de dados de terceiros ou do [Amazon Security Lake](#). A AWS recomenda que você capture os dados de segurança das Regiões da AWS que estão [habilitadas](#) em suas contas, mesmo quando não usadas ativamente.
3. Configure a publicação da fonte de dados nos locais padronizados
 - a. Identifique as fontes de dados de segurança e configure-as para serem publicadas nos locais padronizados. Avalie as opções para exportar dados automaticamente no formato desejado, em vez daquelas em que é necessário desenvolver processos de ETL. Com o Amazon Security Lake, você pode [coletar dados](#) de fontes da AWS compatíveis e de sistemas integrados de terceiros.

4. Configure ferramentas para acessar os locais padronizados
 - a. Configure ferramentas (como o Amazon Athena e o Amazon QuickSight) ou soluções de terceiros para ter o acesso necessário aos locais padronizados. Configure essas ferramentas para operar fora da conta de ferramentas de segurança com acesso de leitura entre contas à conta de arquivo de logs quando aplicável. [Crie assinantes no Amazon Security Lake](#) para fornecer acesso aos dados a essas ferramentas.

Recursos

Práticas recomendadas relacionadas:

- [SEC01-BP01 Separar as workloads usando contas](#)
- [SEC07-BP04 Definir o gerenciamento escalável do ciclo de vida dos dados](#)
- [SEC08-BP04 Impor o controle de acesso](#)
- [OPS08-BP02 Analisar logs de workloads](#)

Documentos relacionados:

- [AWS Whitepapers: Organizing Your AWS Environment Using Multiple Accounts](#)
- [AWS Prescriptive Guidance: AWS Security Reference Architecture \(AWS SRA\)](#)
- [AWS Prescriptive Guidance: Logging and monitoring guide for application owners](#)

Exemplos relacionados:

- [Aggregating, searching, and visualizing log data from distributed sources with Amazon Athena and Amazon QuickSight](#)
- [How to visualize Amazon Security Lake findings with Amazon QuickSight](#)
- [Generate AI powered insights for Amazon Security Lake using Amazon SageMaker Studio and Amazon Bedrock](#)
- [Identify cybersecurity anomalies in your Amazon Security Lake data using Amazon SageMaker](#)
- [Ingest, transform, and deliver events published by Amazon Security Lake to Amazon OpenSearch Service](#)
- [How to use AWS Security Hub and Amazon OpenSearch Service for SIEM](#)

Ferramentas relacionadas:

- [Amazon Security Lake](#)
- [Integrações de parceiros do Amazon Security Lake](#)
- [Open Cybersecurity Schema Framework \(OCSF\)](#)
- [Amazon Athena](#)
- [Amazon QuickSight](#)
- [Amazon Bedrock](#)

SEC04-BP03 Correlacionar e enriquecer os alertas de segurança

Atividades inesperadas podem gerar vários alertas de segurança de diferentes fontes, exigindo maior correlação e enriquecimento para entender todo o contexto. Automatize a correlação e o enriquecimento dos alertas de segurança para ajudar a aumentar a precisão da identificação e resposta aos incidentes.

Resultado desejado: à medida que a atividade gera diferentes alertas em seus ambientes e workloads, mecanismos automatizados correlacionam dados e os enriquecem com informações adicionais. Esse pré-processamento apresenta uma compreensão mais detalhada do evento, o que ajuda os investigadores a determinar a importância do evento e se ele constitui um incidente que requer uma resposta formal. Esse processo reduz a carga sobre suas equipes de monitoramento e investigação.

Antipadrões comuns:

- Diferentes grupos de pessoas investigam descobertas e alertas gerados por sistemas diferentes, a menos que seja exigido de outra forma pelos requisitos de separação de deveres.
- Sua organização canaliza todos os dados de detecção e alerta de segurança para locais padrão, mas exige que os investigadores realizem a correlação e o enriquecimento manualmente.
- Você depende exclusivamente da inteligência dos sistemas de detecção de ameaças para relatar descobertas e determinar a gravidade.

Benefícios do estabelecimento desta prática recomendada: a automatização da correlação e do enriquecimento de alertas ajuda a reduzir o trabalho cognitivo geral e a preparação manual de dados que os investigadores precisam realizar. Essa prática pode reduzir o tempo necessário para

determinar se o evento representa um incidente e iniciar uma resposta formal. O contexto adicional também ajuda a avaliar com precisão a verdadeira gravidade de um evento, pois ela pode ser maior ou menor do que o sugerido por qualquer alerta.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: baixo

Orientações para a implementação

Os alertas de segurança podem vir de várias fontes diferentes na AWS, incluindo:

- Serviços como [Amazon GuardDuty](#), [AWS Security Hub](#), [Amazon Macie](#), [Amazon Inspector](#), [AWS Config](#), [AWS Identity and Access Management Access Analyzer](#) e [Network Access Analyzer](#).
- Alertas da análise automatizada de logs de aplicação, infraestrutura e serviços da AWS, como da [Análise de Segurança para o Amazon OpenSearch Service](#).
- Alarmes em resposta a mudanças em sua atividade de cobrança de fontes como o [Amazon CloudWatch](#), o [Amazon EventBridge](#) ou o [AWS Budgets](#).
- Fontes de terceiros, como feeds de inteligência contra ameaças e [soluções de parceiros de segurança](#) da AWS Partner Network.
- [Contatos feitos pela Equipe de Confiança e Segurança da AWS](#) ou por outras fontes, como clientes ou funcionários internos.

Em sua forma mais fundamental, os alertas contêm informações sobre quem (a entidade principal ou identidade) está fazendo o quê (a ação realizada) para quem (os recursos afetados). Em cada uma dessas fontes, identifique se há maneiras de criar associações nos identificadores referentes a essas identidades, ações e recursos como base para realizar a correlação. Isso poderia ser integrar fontes de alerta a uma ferramenta de gerenciamento de eventos e informações de segurança (SIEM) para realizar a correlação automatizada para você, criar seus próprios pipelines e processamento de dados ou uma combinação de ambos.

Um exemplo de um serviço que pode realizar a correlação para você é o [Amazon Detective](#). O Detective realiza a ingestão contínua de alertas de várias fontes da AWS e de terceiros e usa diferentes formas de inteligência com o objetivo de montar um grafo visual das respectivas relações para auxiliar nas investigações.

Embora a gravidade inicial de um alerta ajude na priorização, o contexto em que o alerta aconteceu determina sua verdadeira gravidade. Como exemplo, o Amazon GuardDuty pode alertar que uma instância do Amazon EC2 dentro da workload está consultando um nome de domínio inesperado. O

GuardDuty pode atribuir um baixo nível de gravidade a esse alerta por si só. Entretanto, a correlação automatizada com outras atividades em torno do momento do alerta pode revelar que várias centenas de instâncias do EC2 foram implantadas pela mesma identidade, o que aumenta os custos operacionais gerais. Nesse caso, o GuardDuty pode publicar esse contexto de evento correlacionado como um novo alerta de segurança e definir a gravidade como alta, o que agilizará ações futuras.

Etapas da implementação

1. Identifique fontes de informações sobre alertas de segurança. Entenda como os alertas desses sistemas representam identidade, ação e recursos para determinar onde a correlação é possível.
2. Estabeleça um mecanismo para capturar alertas de diferentes fontes. Considere serviços como o Security Hub, o EventBridge e o CloudWatch para essa finalidade.
3. Identifique fontes para correlação e enriquecimento de dados. As fontes de exemplo incluem o CloudTrail, logs de fluxo da VPC, o Amazon Security Lake e logs de infraestrutura e aplicação.
4. Integre os alertas às fontes de correlação e enriquecimento de dados para criar contextos de eventos de segurança mais detalhados e determinar a gravidade.
 - a. O Amazon Detective, ferramentas de SIEM ou outras soluções de terceiros podem realizar determinado nível de ingestão, correlação e enriquecimento automaticamente.
 - b. Você também pode usar serviços da AWS para criar seus próprios alertas. Por exemplo, você pode invocar uma função do AWS Lambda para executar uma consulta do Amazon Athena no AWS CloudTrail ou no Amazon Security Lake e publicar os resultados no EventBridge.

Recursos

Práticas recomendadas relacionadas:

- [SEC10-BP03 Preparar recursos forenses](#)
- [OPS08-BP04 Criar alertas acionáveis](#)
- [REL06-BP03 Enviar notificações \(processamento e emissão de alarmes em tempo real\)](#)

Documentos relacionados:

- [AWS Security Incident Response Guide](#)

Exemplos relacionados:

- [How to enrich AWS Security Hub findings with account metadata](#)
- [How to use AWS Security Hub and Amazon OpenSearch Service for SIEM](#)

Ferramentas relacionadas:

- [Amazon Detective](#)
- [Amazon EventBridge](#)
- [AWS Lambda](#)
- [Amazon Athena](#)

SEC04-BP04 Iniciar a correção de recursos irregulares

Seus controles de detecção podem emitir alertas sobre recursos que não estão em conformidade com seus requisitos de configuração. Você pode iniciar correções definidas de maneira programática, tanto manual quanto automaticamente, para corrigir esses recursos e ajudar a minimizar possíveis impactos. Ao definir correções programaticamente, você pode tomar medidas rápidas e consistentes.

Embora a automação possa aprimorar as operações de segurança, você deve implementar e gerenciar a automação com cuidado. Estabeleça mecanismos apropriados de supervisão e controle para verificar se as respostas automatizadas são eficazes e precisas e estão alinhadas com as políticas organizacionais e a propensão ao risco.

Resultado desejado: você define os padrões de configuração de recursos e as etapas de correção para quando forem detectados recursos irregulares. Sempre que possível, você procura definir as correções programaticamente para que elas possam ser iniciadas de modo manual ou por meio de automação. Existem sistemas de detecção para identificar recursos irregulares e publicar alertas em ferramentas centralizadas que são monitoradas por suas equipes de segurança. Essas ferramentas comportam a execução das correções programáticas, tanto manual quanto automaticamente. As correções automáticas têm mecanismos apropriados de supervisão e controle para governar o respectivo uso.

Antipadrões comuns:

- Você implementa a automação, mas não consegue testar e validar minuciosamente as ações de correção. Isso pode resultar em consequências indesejadas, como interrupção de operações comerciais legítimas ou instabilidade no sistema.

- Você melhora os tempos de resposta e os procedimentos por meio da automação, mas sem monitoramento e mecanismos adequados que permitam a intervenção e avaliação humanas quando necessário.
- Você depende exclusivamente de correções, em vez de tê-las como parte de um programa mais amplo de resposta e recuperação de incidentes.

Benefícios do estabelecimento desta prática recomendada: as correções automáticas podem responder a configurações incorretas mais rapidamente do que os processos manuais, o que ajuda a minimizar possíveis impactos nos negócios e reduzir a janela de oportunidades para usos indesejados. Quando você define as correções programaticamente, elas são aplicadas de forma consistente, o que reduz o risco de erro humano. A automação também pode lidar com um volume maior de alertas de forma simultânea, o que é particularmente importante em ambientes que operam em grande escala.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Conforme descrito em [SEC01-BP03 Identificar e validar objetivos de controle](#), serviços como o [AWS Config](#) podem ajudar você a monitorar a configuração dos recursos em suas contas com relação à adesão aos seus requisitos. Quando forem detectados recursos irregulares, é recomendável configurar o envio de alertas a uma solução de gerenciamento de procedimentos de segurança na nuvem (CSPM) (por ex., o [AWS Security Hub](#)) para ajudar na correção. Essas soluções fornecem um local central para os investigadores de segurança monitorarem os problemas e tomarem medidas corretivas.

Embora algumas situações em que há recursos irregulares sejam únicas e exijam avaliação humana para serem corrigidas, outras têm uma resposta padrão que você pode definir programaticamente. Por exemplo, uma resposta padrão a um grupo de segurança da VPC configurado incorretamente pode ser remover as regras não permitidas e notificar o responsável. As respostas podem ser definidas em funções do [AWS Lambda](#), em documentos do [AWS Systems Manager Automation](#) ou por meio de outros ambientes de sua preferência. O ambiente deve estar apto a se autenticar na AWS usando um perfil do IAM com as permissões mínimas necessárias para tomar medidas corretivas.

Após a definição da correção desejada, você poderá determinar seus meios de preferência para iniciá-la. O AWS Config pode [iniciar as correções](#) para você. Se você estiver usando o Security

Hub, poderá fazer isso por meio de [ações personalizadas](#), as quais publicam as informações de descoberta no [Amazon EventBridge](#). Uma regra do EventBridge pode então iniciar a correção. Você pode configurar a ação personalizada no Security Hub para ser executada de forma automática ou manual.

Para correção programática, recomendamos que você tenha logs e auditorias abrangentes das ações realizadas, bem como dos respectivos resultados. Revise e analise esses logs para avaliar a eficácia dos processos automatizados e identificar áreas de melhoria. Capture logs no [Amazon CloudWatch Logs](#) e resultados de correções como [notas de descoberta](#) no Security Hub.

Como ponto de partida, considere a [resposta automatizada de segurança na AWS](#), que tem correções predefinidas para resolver erros comuns de configuração de segurança.

Etapas da implementação

1. Analise e priorize os alertas.
 - a. Consolide alertas de segurança de vários serviços da AWS no Security Hub para ter visibilidade, priorização e correção centralizadas.
2. Desenvolva correções.
 - a. Use serviços como o Systems Manager e o AWS Lambda para executar correções programáticas.
3. Configure como as correções são iniciadas.
 - a. Usando o Systems Manager, defina ações personalizadas que publiquem as descobertas no EventBridge. Configure essas ações para serem iniciadas manual ou automaticamente.
 - b. Você também pode usar o [Amazon Simple Notification Service \(SNS\)](#) para enviar notificações e alertas às partes interessadas relevantes (como equipe de segurança ou equipes de resposta a incidentes) para intervenção manual ou encaminhamento, se necessário.
4. Revise e analise os logs de correção em prol da eficácia e melhoria.
 - a. Envie a saída de logs ao CloudWatch Logs. Capture resultados como notas de descoberta no Security Hub.

Recursos

Práticas recomendadas relacionadas:

- [SEC06-BP03 Reduzir o gerenciamento manual e o acesso interativo](#)

Documentos relacionados:

- [AWS Security Incident Response Guide - Detection](#)

Exemplos relacionados:

- [Automated Security Response on AWS](#)
- [Monitor EC2 instance key pairs using AWS Config](#)
- [Create AWS Config custom rules by using AWS CloudFormation Guard policies](#)
- [Automatically remediate unencrypted Amazon RDS DB instances and clusters](#)

Ferramentas relacionadas:

- [AWS Systems Manager Automation](#)
- [Automated Security Response on AWS](#)

Proteção de infraestrutura

A proteção da infraestrutura abrange metodologias de controle, como defesa em profundidade, que são necessárias para executar melhores práticas e obrigações organizacionais ou regulatórias. O uso dessas metodologias é fundamental para o êxito de operações contínuas na nuvem.

A proteção da infraestrutura é uma parte essencial de um programa de segurança da informação. Ela garante que os sistemas e os serviços de sua carga de trabalho sejam protegidos contra acesso não intencional e não autorizado e possíveis vulnerabilidades. Por exemplo, você definirá limites de confiança (por exemplo, limites de rede e conta), configuração e manutenção da segurança do sistema (por exemplo, fortalecimento, minimização e aplicação de patches), autenticação e autorizações do sistema operacional (por exemplo, usuários, chaves e níveis de acesso) e outros pontos de aplicação de políticas apropriados (por exemplo, firewalls de aplicativos Web e/ou gateways de API).

Regiões, zonas de disponibilidade, zonas locais da AWS e AWS Outposts

É necessário estar familiarizado com regiões, zonas de disponibilidade, [zonas locais da AWS](#) e aos [AWS Outposts](#), que são componentes da infraestrutura global segura da AWS.

A AWS tem o conceito de região, que é um local físico em algum lugar do mundo onde os datacenters são agrupados. Os grupos de datacenters lógicos são chamados de zona de disponibilidade (AZ). Cada região da AWS consiste em várias AZs isoladas e fisicamente separadas em uma área geográfica. Se você tiver requisitos de residência de dados, será possível escolher a região da AWS mais próxima do local desejado. Você mantém controle e propriedade totais sobre a região em que seus dados estão localizados fisicamente, o que pode ser útil para atender aos requisitos regionais de conformidade e residência de dados. Cada AZ tem fontes de energia, refrigeração e segurança física independentes. Se uma aplicação for particionada em AZs, você estará mais bem isolado e protegido contra problemas como queda de energia, raios, tornados, terremotos, entre outros. As AZs estão separadas fisicamente por uma distância significativa, a muitos quilômetros de qualquer outra AZ, embora todas estejam a 100 km (60 milhas) uma da outra. Todas as AZs em uma região da AWS são interconectadas com rede de alta largura de banda e baixa latência, usando fibra metropolitana dedicada totalmente redundante, fornecendo rede de alto throughput e baixa latência entre as AZs. Todo tráfego entre AZs é criptografado. Os clientes da AWS focados em alta disponibilidade podem projetar suas aplicações para serem executadas em várias AZs para obter uma tolerância a falhas ainda maior. As regiões da AWS atendem aos mais altos níveis de segurança, conformidade e proteção de dados.

As zonas locais da AWS colocam computação, armazenamento, banco de dados e outros serviços selecionados da AWS mais próximos dos usuários finais. Com as zonas locais da AWS, é possível executar facilmente aplicações altamente exigentes que requerem latências de milissegundos de um dígito para os usuários finais, como criação de conteúdo de mídia e entretenimento, jogos em tempo real, simulações de reservatórios, automação de projetos eletrônicos e machine learning. Cada zona local da AWS é uma extensão de uma região da AWS onde é possível executar as simulações de reservatórios sensíveis à latência, usando serviços da AWS como Amazon EC2, Amazon VPC, Amazon EBS, Amazon File Storage e Elastic Load Balancing na proximidade geográfica dos usuários finais. As zonas locais da AWS fornecem uma conexão segura e de alta largura de banda entre as workloads locais e aquelas executadas na região da AWS, permitindo se conectar perfeitamente a toda a gama de serviços na região por meio das mesmas APIs e conjuntos de ferramentas.

O AWS Outposts inclui serviços nativos, infraestrutura e modelos operacionais da AWS para praticamente qualquer datacenter, espaço de colocação ou instalação on-premises. Você pode usar as mesmas APIs, ferramentas e infraestrutura da AWS em instalações on-premises e na Nuvem AWS para oferecer uma experiência híbrida verdadeiramente consistente. O AWS Outposts foi projetado para ambientes conectados e pode ser usado para oferecer suporte a workloads que devem permanecer no ambiente on-premises devido à baixa latência ou às necessidades de processamento de dados locais.

A AWS tem várias abordagens para a proteção da infraestrutura. As seções a seguir descrevem como usar essas abordagens.

Tópicos

- [Proteção de redes](#)
- [Proteção da computação](#)

Proteção de redes

Os usuários, tanto a sua força de trabalho quanto seus clientes, podem estar em qualquer lugar. Você precisa se afastar dos modelos tradicionais que confiam em qualquer pessoa e em tudo que tenha acesso à sua rede. Ao seguir o princípio de aplicar segurança em todas as camadas, você emprega uma abordagem de [confiança zero](#). A segurança de confiança zero é um modelo em que os componentes da aplicação ou microsserviços são considerados distintos uns dos outros e nenhum componente ou microsserviço confia em outro.

O planejamento e o gerenciamento minuciosos do design da rede são a base do isolamento e dos limites para os recursos em sua carga de trabalho. Como muitos recursos da carga de trabalho operam em uma VPC e herdam as propriedades de segurança, é essencial que o projeto tenha o suporte de mecanismos de inspeção e proteção respaldados por automação. Da mesma forma, para cargas de trabalho que operam fora de uma VPC, usando serviços puramente de borda e/ou sem servidor, as melhores práticas se aplicam em uma abordagem mais simplificada. Consulte o [AWS Well-Architected Serverless Applications Lens \(Lentes de aplicações sem servidor do AWS Well-Architected\)](#) para obter orientações específicas sobre a segurança sem servidor.

Práticas recomendadas

- [SEC05-BP01 Criar camadas de rede](#)
- [SEC05-BP02 Controlar o fluxo de tráfego dentro das camadas de rede](#)
- [SEC05-BP03 Implementar proteção baseada em inspeção](#)
- [SEC05-BP04 Automatizar a proteção da rede](#)

SEC05-BP01 Criar camadas de rede

Segmente a topologia de rede em diferentes camadas com base nos agrupamentos lógicos dos componentes da workload e de acordo com a confidencialidade dos dados e os requisitos de acesso. Diferencie os componentes que exigem acesso de entrada pela internet, como endpoints públicos da web, e aqueles que precisam apenas de acesso interno, como bancos de dados.

Resultado desejado: as camadas de sua rede fazem parte de uma abordagem de segurança de defesa profunda integral que complementa a estratégia de autenticação e autorização de identidade de suas workloads. As camadas estão posicionadas de acordo com a confidencialidade dos dados e os requisitos de acesso, com mecanismos apropriados de fluxo e controle de tráfego.

Antipadrões comuns:

- Você cria todos os recursos em uma única VPC ou sub-rede.
- Você constrói as camadas de rede sem considerar os requisitos de confidencialidade dos dados, o comportamento dos componentes ou a funcionalidade.
- Você usa VPCs e sub-redes como padrão para todas as considerações de camada de rede e não considera como os serviços gerenciados da AWS influenciam sua topologia.

Benefícios do estabelecimento desta prática recomendada: estabelecer camadas de rede é a primeira etapa para restringir caminhos desnecessários pela rede, especialmente aqueles que levam a sistemas e dados críticos. Desse modo, o acesso de agentes não autorizados à sua rede e a outros recursos dentro dela torna-se mais difícil. Camadas de rede discretas reduzem de forma favorável o escopo da análise para sistemas de inspeção (por ex., para detecção de intrusões ou prevenção de malware). Isso reduz a possibilidade de falsos positivos e a sobrecarga de processamento desnecessária.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

Quando se projeta uma arquitetura de workload, é comum separar os componentes em diferentes camadas com base nas respectivas responsabilidades. Por exemplo, uma aplicação web pode ter uma camada de apresentação, uma camada de aplicação e uma camada de dados. Você pode adotar uma abordagem semelhante ao projetar sua topologia de rede. Os controles de rede subjacentes podem ajudar a aplicar os requisitos de acesso aos dados da workload. Por exemplo, em uma arquitetura de aplicação web de três camadas, é possível armazenar os arquivos da camada de apresentação estática no [Amazon S3](#) e servi-los por meio de uma rede de entrega de conteúdo (CDN), como o [Amazon CloudFront](#). A camada de aplicação pode ter endpoints públicos servidos por um [Application Load Balancer \(ALB\)](#) em uma sub-rede pública da [Amazon VPC](#) [de modo semelhante a uma zona desmilitarizada (DMZ)], com serviços de back-end implantados em sub-redes privadas. A camada de dados, que hospeda recursos como bancos de dados e sistemas de arquivos compartilhados, pode residir em diferentes sub-redes privadas dos recursos da camada de aplicação. Em cada um desses limites de camada (CDN, sub-rede pública, sub-rede privada), você pode implantar controles que permitam somente a entrada do tráfego autorizado.

De modo semelhante à modelagem de camadas de rede com base na finalidade funcional dos componentes da workload, considere também a confidencialidade dos dados que estão sendo processados. Usando o exemplo de aplicação web, embora todos os serviços de workload possam residir na camada de aplicação, serviços diferentes podem processar dados com diferentes níveis de confidencialidade. Nesse caso, dividir a camada de aplicação usando várias sub-redes privadas, diferentes VPCs na mesma Conta da AWS ou até mesmo diferentes VPCs em diferentes Contas da AWS para cada nível de confidencialidade de dados pode ser apropriado de acordo com seus requisitos de controle.

Uma consideração adicional sobre as camadas de rede é a consistência do comportamento dos componentes da workload. Continuando com o exemplo, na camada de aplicação, você pode ter

serviços que aceitem entradas de usuários finais ou integrações de sistemas externos que são inerentemente mais arriscadas do que as entradas de outros serviços. Os exemplos incluem uploads de arquivos, scripts de código para execução, verificação de e-mails e assim por diante. Colocar esses serviços em uma camada de rede própria ajuda a criar um limite de isolamento mais forte em torno deles e pode impedir que o comportamento exclusivo de cada um deles crie alertas falsos positivos nos sistemas de inspeção.

Como parte do seu design, considere como o uso de serviços gerenciados da AWS influencia sua topologia de rede. Examine de que forma serviços como o [Amazon VPC Lattice](#) podem ajudar a facilitar a interoperabilidade dos componentes da workload nas camadas da rede. Ao usar o [AWS Lambda](#), implante as sub-redes da VPC, a menos que haja motivos específicos para não o fazer. Determine onde os endpoints da VPC e o [AWS PrivateLink](#) podem simplificar a adesão às políticas de segurança que limitam o acesso aos gateways da internet.

Etapas da implementação

1. Analise sua arquitetura de workload. Agrupe logicamente os componentes e serviços com base nas funções às quais eles atendem, na confidencialidade dos dados que estão sendo processados e no respectivo comportamento.
2. Com relação a componentes que respondem a solicitações da internet, considere usar balanceadores de carga ou outros proxies para fornecer endpoints públicos. Examine as alterações nos controles de segurança usando serviços gerenciados, como o CloudFront, o [Amazon API Gateway](#), o Elastic Load Balancing e o [AWS Amplify](#), para hospedar endpoints públicos.
3. No caso de componentes executados em ambientes de computação, como instâncias do Amazon EC2, contêineres do [AWS Fargate](#) ou funções do Lambda, implante-os em sub-redes privadas com base em seus grupos desde a primeira etapa.
4. Em serviços da AWS totalmente gerenciados, como o [Amazon DynamoDB](#), o [Amazon Kinesis](#) ou o [Amazon SQS](#), considere a possibilidade de usar endpoints da VPC como padrão para acesso via endereços IP privados.

Recursos

Práticas recomendadas relacionadas:

- [REL02 Planejar a topologia de rede](#)
- [PERF04-BP01 Compreender como as redes afetam a performance](#)

Vídeos relacionados:

- [AWS re:Invent 2023 - AWS networking foundations](#)

Exemplos relacionados:

- [Exemplos de VPC](#)
- [Access container applications privately on Amazon ECS by using AWS Fargate, AWS PrivateLink, and a Network Load Balancer](#)
- [Serve static content in an Amazon S3 bucket through a VPC by using Amazon CloudFront](#)

SEC05-BP02 Controlar o fluxo de tráfego dentro das camadas de rede

Dentro das camadas da sua rede, use uma segmentação adicional para restringir o tráfego somente aos fluxos necessários para cada workload. Primeiro, concentre-se em controlar o tráfego entre a internet ou outros sistemas externos e uma workload e o seu ambiente (tráfego norte-sul). Depois, observe os fluxos entre os diferentes componentes e sistemas (tráfego leste-oeste).

Resultado desejado: somente os fluxos de rede necessários são permitidos para que os componentes das workloads comuniquem-se entre si e com os clientes e com quaisquer outros serviços dos quais dependam. Seu design considera questões como comparação entre entradas e saídas públicas e privadas, classificação de dados, regulamentações regionais e requisitos de protocolo. Sempre que possível, você prefere fluxos ponto a ponto em vez de emparelhamento de rede como parte do princípio de privilégio mínimo.

Antipadrões comuns:

- Você adota uma abordagem de segurança de rede baseada em perímetro e controla apenas o fluxo de tráfego no limite das camadas de sua rede.
- Você presume que todo o tráfego dentro de uma camada de rede está autenticado e autorizado.
- Você aplica controles para o tráfego de entrada ou de saída, mas não para ambos.
- Você depende exclusivamente dos componentes da workload e dos controles de rede para autenticar e autorizar o tráfego.

Benefícios do estabelecimento desta prática recomendada: esta prática ajuda a reduzir o risco de movimentação não autorizada em sua rede e adiciona uma camada extra de autorização às

workloads. Ao realizar o controle do fluxo de tráfego, você pode restringir o escopo do impacto de um incidente de segurança e acelerar a detecção e a resposta.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

Embora as camadas de rede ajudem a estabelecer os limites em torno de componentes da workload que atendem a funções, níveis de confidencialidade de dados e comportamentos semelhantes, você pode criar um nível de controle de tráfego bem mais refinado usando técnicas para segmentar ainda mais os componentes dentro dessas camadas, seguindo o princípio de privilégio mínimo. Na AWS, as camadas de rede são definidas principalmente usando sub-redes de acordo com os intervalos de endereços IP dentro de uma Amazon VPC. As camadas também podem ser definidas usando diferentes VPCs, como para agrupar ambientes de microsserviços por domínio de negócios. Ao usar várias VPCs, use o [AWS Transit Gateway](#) para mediar o roteamento. Embora isso forneça controle de tráfego no nível da camada 4 (endereços IP e intervalos de portas) usando grupos de segurança e tabelas de rotas, você pode obter maior controle usando serviços adicionais como o [AWS PrivateLink](#), o [Firewall de DNS do Amazon Route 53 Resolver](#), o [AWS Network Firewall](#) e o [AWS WAF](#).

Entenda e faça um inventário do fluxo de dados e dos requisitos de comunicação das workloads no que se refere a partes de conexão e iniciação, portas, protocolos e camadas de rede. Avalie os protocolos disponíveis para estabelecer conexões, transmitir dados e selecionar aqueles que atendam aos seus requisitos de proteção (por exemplo, HTTPS em vez de HTTP). Capture esses requisitos nos limites de suas redes e dentro de cada camada. Depois que esses requisitos forem identificados, explore as opções para permitir que apenas o tráfego necessário flua em cada ponto de conexão. Um bom ponto de partida é usar grupos de segurança na VPC, pois eles podem ser anexados a recursos que usam uma interface de rede elástica (ENI), como instâncias do Amazon EC2, tarefas do Amazon ECS, pods do Amazon EKS ou bancos de dados do Amazon RDS. Ao contrário de um firewall de camada 4, um grupo de segurança pode ter uma regra que permite o tráfego de outro grupo de segurança por meio do respectivo identificador, minimizando as atualizações à medida que os recursos dentro do grupo mudam ao longo do tempo. Você também pode filtrar o tráfego por meio de regras de entrada e saída usando grupos de segurança.

Quando o tráfego se move entre as VPCs, é comum usar o emparelhamento de VPCs para roteamento simples ou o AWS Transit Gateway para roteamento complexo. Com essas abordagens, você facilita os fluxos de tráfego entre o intervalo de endereços IP das redes de origem e de destino. No entanto, se a workload exigir apenas fluxos de tráfego entre componentes específicos em

diferentes VPCs, considere usar uma conexão ponto a ponto por meio do [AWS PrivateLink](#). Para isso, identifique qual serviço deve atuar como produtor e qual deve atuar como consumidor. Implante um balanceador de carga compatível para o produtor, ative o PrivateLink adequadamente e aceite uma solicitação de conexão do consumidor. Em seguida, o serviço do produtor recebe um endereço IP privado da VPC do consumidor, que este pode usar para fazer solicitações subsequentes. Essa abordagem reduz a necessidade de emparelhar as redes. Inclua os custos de processamento de dados e balanceamento de carga como parte da avaliação do PrivateLink.

Embora os grupos de segurança e o PrivateLink ajudem a controlar o fluxo entre os componentes das workloads, outra consideração importante é como controlar quais domínios DNS os recursos podem acessar (se houver). Dependendo da configuração DHCP das VPCs, você pode considerar dois diferentes serviços da AWS para essa finalidade. A maioria dos clientes usa o serviço de DNS do Route 53 Resolver padrão (também chamado de servidor de DNS da Amazon ou AmazonProvidedDNS) disponível para VPCs no endereço +2 do respectivo intervalo CIDR. Com essa abordagem, você pode criar regras de firewall de DNS e associá-las à VPC para determinar quais ações devem ser realizadas para as listas de domínios que você fornece.

Se você não estiver usando o Route 53 Resolver, ou se quiser complementar o Resolver com recursos mais detalhados de inspeção e controle de fluxo, além da filtragem de domínio, considere implantar um AWS Network Firewall. Esse serviço inspeciona pacotes individuais usando regras sem estado ou com estado para determinar se deve negar ou permitir o tráfego. Você pode adotar uma abordagem semelhante para filtrar o tráfego de entrada da web para seus endpoints públicos usando o AWS WAF. Para obter mais orientações sobre esses serviços, consulte [SEC05-BP03 Implementar proteção baseada em inspeção](#).

Etapas da implementação

1. Identifique os fluxos de dados necessários entre os componentes das workloads.
2. Aplique vários controles com uma abordagem de defesa profunda para tráfego de entrada e saída, incluindo o uso de grupos de segurança e tabelas de rotas.
3. Use firewalls para definir um controle refinado sobre o tráfego de rede que entra, sai e atravessa suas VPCs, como o Firewall de DNS do Route 53 Resolver, o AWS Network Firewall e o AWS WAF. Considere usar o [AWS Firewall Manager](#) para configurar e gerenciar centralmente as regras de firewall em toda a organização.

Recursos

Práticas recomendadas relacionadas:

- [REL03-BP01 Escolher como segmentar a workload](#)
- [SEC09-BP02 Aplicar a criptografia em trânsito](#)

Documentos relacionados:

- [Security best practices for your VPC](#)
- [AWS Network Optimization Tips](#)
- [Guidance for Network Security on AWS](#)
- [Secure your VPC's outbound network traffic in the Nuvem AWS](#)

Ferramentas relacionadas:

- [AWS Firewall Manager](#)

Vídeos relacionados:

- [AWS Transit Gateway reference architectures for many VPCs](#)
- [Application Acceleration and Protection with Amazon CloudFront, AWS WAF, and AWS Shield](#)
- [AWS re:Inforce 2023: Firewalls and where to put them](#)

Exemplos relacionados:

- [Laboratório: CloudFront for Web Application](#)

SEC05-BP03 Implementar proteção baseada em inspeção

Configure pontos de inspeção de tráfego entre as camadas de rede para garantir que os dados em trânsito correspondam aos padrões e categorias esperados. Analise padrões, metadados e fluxos de tráfego para ajudar a identificar, detectar e responder a eventos com maior eficiência.

Resultado desejado: o tráfego que atravessa as camadas da rede é inspecionado e autorizado. As decisões de permissão e negação baseiam-se em regras explícitas, inteligência contra ameaças e desvios dos comportamentos de referência. As proteções tornam-se mais rígidas à medida que o tráfego aproxima-se dos dados confidenciais.

Antipadrões comuns:

- Confiar somente em regras de firewall baseadas em portas e protocolos. Não aproveitar os sistemas inteligentes.
- Criar regras de firewall com base em padrões específicos de ameaças atuais que estão sujeitos a alterações.
- Inspeccionar somente o tráfego que transita de sub-redes privadas para públicas ou de sub-redes públicas para a internet.
- Não ter uma visão de referência do tráfego da rede para comparar com anomalias de comportamento.

Benefícios do estabelecimento desta prática recomendada: os sistemas de inspeção permitem que você crie regras inteligentes, como permitir ou negar tráfego somente quando existem determinadas condições nos dados de tráfego. Beneficie-se de conjuntos de regras gerenciados pela AWS e por parceiros, com base na inteligência contra ameaças mais recente, à medida que o cenário de ameaças muda ao longo do tempo. Isso reduz as despesas indiretas de manter regras e pesquisar indicadores de comprometimento, reduzindo o potencial de falsos positivos.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Tenha controle refinado sobre o tráfego de rede com estado e sem estado usando o AWS Network Firewall ou outros [firewalls](#) e os [sistemas de prevenção de invasões \(IPS\)](#) no AWS Marketplace que você pode implantar por trás de um Gateway Load Balancer (GWLB). O AWS Network Firewall comporta especificações de IPS de código aberto [compatíveis com o Suricata](#) para ajudar a proteger as workloads.

Tanto o AWS Network Firewall quanto as soluções de fornecedor que usam um GWLB comportam diferentes modelos de implantação de inspeção em linha. Por exemplo, você pode realizar a inspeção por VPC, centralizar em uma VPC de inspeção ou implantar em um modelo híbrido em que o tráfego leste-oeste flui por meio de uma VPC de inspeção e a entrada da internet é inspecionada por VPC. Outra consideração é se a solução comporta o desempacotamento do Transport Layer Security (TLS), permitindo a inspeção detalhada de pacotes para fluxos de tráfego iniciados em qualquer direção. Para obter mais informações e detalhes sobre essas configurações, consulte o [Guia de práticas recomendadas da AWS Network Firewall](#).

Se você estiver usando soluções que realizam inspeções fora de banda, como análise de pcap de dados de pacotes de interfaces de rede operando no modo promíscuo, você pode configurar o

[espelhamento de tráfego da VPC](#). O tráfego espelhado é computado na largura de banda disponível de suas interfaces e está sujeito às mesmas cobranças de transferência de dados que o tráfego não espelhado. Você pode ver se as versões virtuais desses dispositivos estão disponíveis no [AWS Marketplace](#), o qual pode comportar a implantação em linha por trás de um GWLB.

Com relação a componentes executados por meio de protocolos baseados em HTTP, proteja sua aplicação contra ameaças comuns com um firewall de aplicações web (WAF). O [AWS WAF](#) é um firewall de aplicações web que permite monitorar e bloquear solicitações HTTP(S) que correspondam às suas regras configuráveis antes de enviá-las ao Amazon API Gateway, ao Amazon CloudFront, ao AWS AppSync ou a um Application Load Balancer. Considere a inspeção detalhada de pacotes ao avaliar a implantação do firewall de aplicações web, pois alguns exigem que você encerre o TLS antes da inspeção de tráfego. Para começar a utilizar o AWS WAF, você pode usar o [AWS Managed Rules](#) em conjunto com as suas próprias regras ou usar [integrações de parceiros](#) existentes.

Você pode gerenciar centralmente o AWS WAF, AWS Shield Advanced, o AWS Network Firewall e grupos de segurança da Amazon VPC em sua organização da AWS com o [AWS Firewall Manager](#).

Etapas da implementação

1. Determine se você pode definir um escopo amplo das regras de inspeção, como por meio de uma VPC de inspeção, ou se precisa de uma abordagem mais detalhada por VPC.
2. Para soluções de inspeção em linha:
 - a. Se estiver usando o AWS Network Firewall, crie regras, políticas de firewall e o próprio firewall. Depois de configurados, você pode [rotear o tráfego para o endpoint do firewall](#) para permitir a inspeção.
 - b. Se estiver usando um dispositivo de terceiros com um Gateway Load Balancer (GWLB), implante e configure seu dispositivo em uma ou mais zonas de disponibilidade. Em seguida, crie o GWLB, o serviço de endpoint e o endpoint e configure o roteamento para o tráfego.
3. Para soluções de inspeção fora de banda:
 1. Ative o espelhamento de tráfego da VPC em interfaces nas quais o tráfego de entrada e saída deve ser espelhado. Você pode usar regras do Amazon EventBridge para invocar uma função do AWS Lambda que ative o espelhamento de tráfego em interfaces quando são criados recursos. Aponte as sessões de espelhamento de tráfego para o Network Load Balancer na frente do dispositivo que processa o tráfego.
4. Para soluções de tráfego de entrada na web:
 - a. Para configurar o AWS WAF, primeiro configure uma lista de controle de acesso à web (ACL da web). ACL da web é um conjunto de regras com uma ação padrão processada em série

(PERMITIR ou NEGAR) que define como o WAF lida com o tráfego. Você pode criar seus próprios grupos e regras ou usar grupos de regras gerenciadas da AWS em sua ACL da web.

- b. Assim que configurada, a ACL da web pode ser associada a um recurso da AWS (como um Application Load Balancer, uma API REST do API Gateway ou uma distribuição do CloudFront) para começar a proteger o tráfego da web.

Recursos

Documentos relacionados:

- [What is Traffic Mirroring?](#)
- [Implementing inline traffic inspection using third-party security appliances](#)
- [AWS Network Firewall example architectures with routing](#)
- [Centralized inspection architecture with AWS Gateway Load Balancer and AWS Transit Gateway](#)

Exemplos relacionados:

- [Best practices for deploying Gateway Load Balancer](#)
- [TLS inspection configuration for encrypted egress traffic and AWS Network Firewall](#)

Ferramentas relacionadas:

- [AWS Marketplace IDS/IPS](#)

SEC05-BP04 Automatizar a proteção da rede

Automatize a implantação das proteções de rede usando práticas de DevOps, como infraestrutura como código (IaC) e pipelines de CI/CD. Essas práticas podem ajudar você a monitorar alterações nas proteções da rede por meio de um sistema de controle de versão, reduzir o tempo necessário para implantar alterações e detectar se as proteções de rede se desviam da configuração desejada.

Resultado desejado: você define as proteções de rede com modelos e as submete a um sistema de controle de versão. Quando novas alterações são feitas, os pipelines automatizados são iniciados para orquestrar os respectivos testes e a implantação. Verificações de políticas e outros testes estáticos estão em vigor para validar as alterações antes da implantação. Você implanta as alterações em um ambiente de preparação para validar se os controles estão operando conforme

o esperado. A implantação nos ambientes de produção também é executada automaticamente quando os controles são aprovados.

Antipadrões comuns:

- Contar com equipes de workload individuais para que definam sua pilha de rede completa, proteções e automações. Não publicar os aspectos padrão da pilha de rede e das proteções de maneira centralizada para as equipes de workload consumirem.
- Contar com uma equipe de rede central para definir todos os aspectos da rede, proteções e automações. Não delegar aspectos específicos da workload da pilha de rede e das proteções à equipe da workload em questão.
- Conseguir o equilíbrio certo de centralização e delegação entre a equipe de rede e as equipes de workload, mas não aplicar padrões consistentes de teste e implantação aos modelos de IaC e pipelines de CI/CD. Não capturar as configurações necessárias em ferramentas que verificam a aderência aos modelos.

Benefícios do estabelecimento desta prática recomendada: o uso de modelos para definir proteções de rede permite acompanhar e comparar as alterações ao longo do tempo com um sistema de controle de versão. Usar a automação para testar e implantar alterações gera padronização e previsibilidade, aumentando as chances de uma implantação bem-sucedida e reduzindo configurações manuais repetitivas.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Vários controles de proteção de rede descritos em [SEC05-BP02 Controlar o fluxo de tráfego dentro das camadas de rede](#) e em [SEC05-BP03 Implementar proteção baseada em inspeção](#) contam com sistemas de regras gerenciadas que podem ser atualizadas automaticamente com base na inteligência contra ameaças mais recente. Exemplos de proteção de endpoints da web incluem [regras gerenciadas do AWS WAF](#) e [mitigação automática de DDoS na camada de aplicação do AWS Shield Advanced](#). Use [grupos de regras gerenciadas do AWS Network Firewall](#) para se manter a par das listas de domínios de baixa reputação e também das assinaturas de ameaças.

Além das regras gerenciadas, recomendamos usar práticas de DevOps para automatizar a implantação dos recursos de rede, das proteções e das regras que você especificar. Você pode capturar essas definições no [AWS CloudFormation](#) ou em outra ferramenta de infraestrutura como código (IaC) de sua preferência, enviá-las a um sistema de controle de versão e implantá-las

por meio de pipelines de CI/CD. Use essa abordagem para obter os benefícios tradicionais do DevOps para gerenciar os controles de rede, como versões mais previsíveis, testes automatizados com ferramentas como o [AWS CloudFormation Guard](#) e detecção de diferenças entre o ambiente implantado e a configuração desejada.

Com base nas decisões que você tomou como parte de [SEC05-BP01 Criar camadas de rede](#), é possível ter uma abordagem de gerenciamento central para criar VPCs dedicadas aos fluxos de entrada, saída e inspeção. Conforme descrito em [AWS Security Reference Architecture \(AWS SRA\)](#), você pode definir essas VPCs em uma [conta de infraestrutura de rede](#) dedicada. Você pode usar técnicas semelhantes para definir centralmente as VPCs usadas pelas workloads em outras contas, os respectivos grupos de segurança, as implantações do AWS Network Firewall, as regras do Route 53 Resolver e as configurações do Firewall de DNS e outros recursos de rede. Você pode compartilhar esses recursos com suas outras contas com o [AWS Resource Access Manager](#). Com essa abordagem, é possível simplificar os testes automatizados e a implantação dos controles de rede na conta de rede, com apenas um destino para gerenciar. É possível fazer isso em um modelo híbrido, no qual você implanta e compartilha determinados controles centralmente e delega outros controles às equipes individuais de workload e respectivas contas.

Etapas da implementação

1. Estabeleça a propriedade para definir quais aspectos da rede e das proteções são definidos centralmente e quais as equipes de workload podem manter.
2. Crie ambientes para testar e implantar alterações na rede e nas respectivas proteções. Por exemplo, use uma conta de teste de rede e uma conta de produção de rede.
3. Determine como você armazenará e manterá os modelos em um sistema de controle de versão. Os modelos centrais podem ser armazenados em um repositório diferente dos repositórios de workload, ao passo que os modelos de workload podem ser armazenados em repositórios específicos para essa workload.
4. Crie pipelines de CI/CD para testar e implantar modelos. Defina testes para verificar se há configurações incorretas e se os modelos estão de acordo com os padrões da sua empresa.

Recursos

Práticas recomendadas relacionadas:

- [SEC01-BP06 Automatizar a implantação de controles de segurança padrão](#)

Documentos relacionados:

- [AWS Security Reference Architecture - Network account](#)

Exemplos relacionados:

- [AWS Deployment Pipeline Reference Architecture](#)
- [NetDevSecOps to modernize AWS networking deployments](#)
- [Integrating AWS CloudFormation security tests with AWS Security Hub and AWS CodeBuild reports](#)

Ferramentas relacionadas:

- [AWS CloudFormation](#)
- [AWS CloudFormation Guard](#)
- [cfn_nag](#)

Proteção da computação

Os recursos de computação incluem instâncias do EC2, contêineres, funções do AWS Lambda, serviços de banco de dados, dispositivos de IoT e muito mais. Cada um desses tipos de recursos de computação requer abordagens diferentes para protegê-los. No entanto, eles compartilham estratégias comuns que devem ser consideradas: defesa em profundidade, gerenciamento de vulnerabilidades, redução na superfície de ataque, automação de configuração e operação e execução de ações à distância. Nesta seção, você encontrará orientações gerais para proteger seus recursos de computação para os principais serviços. Para cada serviço da AWS usado, é importante verificar as recomendações de segurança específicas na documentação do serviço.

Práticas recomendadas

- [SEC06-BP01 Fazer o gerenciamento de vulnerabilidades](#)
- [SEC06-BP02 Provisionar computação por meio de imagens reforçadas](#)
- [SEC06-BP03 Reduzir o gerenciamento manual e o acesso interativo](#)
- [SEC06-BP04 Validar a integridade do software](#)
- [SEC06-BP05 Automatizar a proteção da computação](#)

SEC06-BP01 Fazer o gerenciamento de vulnerabilidades

Verifique e corrija com frequência vulnerabilidades no código, nas dependências e na infraestrutura para proteger-se contra novas ameaças.

Resultado desejado: criar e manter um programa de gerenciamento de vulnerabilidade. Verificar regularmente e corrigir recursos, como instâncias do Amazon EC2, contêineres do Amazon Elastic Container Service (Amazon ECS) e workloads do Amazon Elastic Kubernetes Service (Amazon EKS). Configurar janelas de manutenção para recursos gerenciados da AWS, como bancos de dados Amazon Relational Database Service (Amazon RDS). Utilizar a verificação de código estático para inspecionar a existência de problemas comuns no código-fonte da aplicação. Considerar testes de penetração de aplicações da web se sua organização tiver as habilidades obrigatórias ou puder contratar assistência externa.

Antipadrões comuns:

- Não ter um programa de gerenciamento de vulnerabilidades.
- Realizar aplicação de patches do sistema sem considerar a gravidade ou como evitar riscos.
- Utilizar software que ultrapassou a data de fim de vida útil (EOL) indicada pelo fornecedor.
- Implantar código em produção antes de analisar a existência de problemas de segurança.

Benefícios do estabelecimento desta prática recomendada:

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientação de implementação

Um programa de gerenciamento de vulnerabilidades inclui avaliação de segurança, identificação de problemas, priorização e realização de operações de patch como parte da solução dos problemas. A automação é a chave para verificar de forma contínua as workloads quanto a problemas e exposição acidental à rede e realização de remediação. Automatizar a criação e atualizar os recursos economiza tempo e reduz o risco de erros de configuração que criam mais problemas. Um programa de gerenciamento de vulnerabilidades bem projetado também deve considerar testes de vulnerabilidades durante o desenvolvimento e os estágios de implantação do ciclo de vida do software. Implementar o gerenciamento de vulnerabilidades durante o desenvolvimento e a implantação ajuda a reduzir a chance de uma vulnerabilidade atingir seu ambiente de produção.

Implementar um programa de gerenciamento de vulnerabilidades exige um bom entendimento do [Modelo de responsabilidade compartilhada da AWS](#) e como ele se relaciona com suas workloads

específicas. Segundo o modelo de responsabilidade compartilhada, a AWS é responsável por proteger a infraestrutura da Nuvem AWS. Essa infraestrutura abrange o hardware, o software, as redes e as instalações que executam os serviços da Nuvem AWS. Você é responsável pela segurança na nuvem, por exemplo, os dados reais, a configuração de segurança e as tarefas de gerenciamento de instâncias do Amazon EC2 e por garantir que seus objetos do Amazon S3 sejam classificados e configurados corretamente. Sua abordagem ao gerenciamento de vulnerabilidades também pode variar dependendo dos serviços consumidos. Por exemplo, a AWS gerencia a aplicação de patches para nosso serviço de banco de dados relacional gerenciado, o Amazon RDS, mas você seria responsável pela colocação de patches em bancos de dados auto-hospedados.

A AWS tem uma série de serviços para ajudar com seu programa de gerenciamento de vulnerabilidades. O [Amazon Inspector](#) verifica de forma contínua as workloads da AWS quanto a problemas de software e acesso acidental à rede. O [AWS Systems Manager Patch Manager](#) ajuda a gerenciar a aplicação de patches em suas instâncias do Amazon EC2. O Amazon Inspector e o Systems Manager podem ser visualizados no [AWS Security Hub](#), um serviço de gerenciamento de procedimentos de segurança na nuvem que ajuda a automatizar verificações de segurança da AWS e centralizar alertas de segurança.

O [Amazon CodeGuru](#) pode ajudar a identificar possíveis problemas em aplicações Java e Python utilizando análise de código estático.

Etapas da implementação

- Configurar o [Amazon Inspector](#): o Amazon Inspector detecta automaticamente instâncias do Amazon EC2 recém-executadas, funções do Lambda e imagens de contêiner elegíveis enviadas ao Amazon ECR e as verifica imediatamente quanto a problemas de software, possíveis defeitos e exposição acidental à rede.
- Verificar o código-fonte: verifique as bibliotecas e as dependências quanto a problemas e defeitos. O [Amazon CodeGuru](#) pode verificar e fornecer recomendações para corrigir [problemas de segurança comuns](#) para aplicações Java e Python. [A OWASP Foundation](#) publica uma lista de ferramentas de análise de código-fonte (também conhecidas como ferramentas SAST).
- Implementar um mecanismo para verificar e aplicar patches ao seu ambiente existente, bem como verificação como parte de um processo de construção de pipeline de CI/CD: implemente um mecanismo para verificar e aplicar patches quanto a problemas em suas dependências e sistemas operacionais a fim de ajudar a proteger-se contra novas ameaças. Execute esse mecanismo regularmente. O gerenciamento de vulnerabilidade de software é essencial ao entendimento de onde é necessário aplicar patches ou resolver problemas de software. Priorize a remediação de possíveis problemas de segurança incorporando avaliações de vulnerabilidade no início de

- seu pipeline de integração/entrega contínua (CI/CD). Sua abordagem pode variar com base nos serviços da AWS que você está consumindo. Para conferir a existência de possíveis problemas no software em execução em instâncias do Amazon EC2, adicione o [Amazon Inspector](#) ao seu pipeline para alertar e interromper o processo de compilação se forem detectados problemas ou possíveis defeitos. O Amazon Inspector monitora recursos de forma contínua. Você também pode utilizar produtos de código aberto, como [OWASP Dependency-Check](#), [Snyk](#), [OpenVAS](#), gerenciadores de pacotes e ferramentas de AWS Partner para gerenciamento de vulnerabilidades.
- Utilize o [AWS Systems Manager](#): você é responsável pelo gerenciamento de patches para seus recursos do AWS, incluindo instâncias do Amazon Elastic Compute Cloud (Amazon EC2), imagens de máquina da Amazon (AMIs) e outros recursos de computação. O [AWS Systems Manager Patch Manager](#) automatiza o processo de aplicação de patches em instâncias gerenciadas com atualizações relacionadas à segurança e outros tipos de atualizações. O Patch Manager pode ser utilizado para aplicar patches em instâncias do Amazon EC2 para sistemas operacionais e aplicações, como aplicações da Microsoft, pacotes de serviços Windows e atualizações de versão secundária para instâncias baseadas em Linux. Além do Amazon EC2, o Patch Manager também pode ser utilizado para aplicar patches em servidores on-premises.

Para ter uma lista de sistemas operacionais compatíveis, consulte [Sistemas operacionais compatíveis](#) no Guia do usuário do Systems Manager. Você pode verificar instâncias para ver apenas um relatório de patches ausentes ou verificar e instalar automaticamente todos os patches ausentes.

- Utilize do [AWS Security Hub](#): o Security Hub oferece uma visão abrangente do estado de seu sistema na AWS. Ele coleta dados de segurança em [vários serviços da AWS](#) e oferece essas descobertas em um formato personalizado, possibilitando priorizar as descobertas de segurança em serviços da AWS.
- Utilize o [AWS CloudFormation](#): o [AWS CloudFormation](#) é um serviço de infraestrutura como código (IaC) que pode ajudar com o gerenciamento de vulnerabilidades automatizando a implantação de recursos e padronizando a arquitetura de recursos em várias contas e ambientes.

Recursos

Documentos relacionados:

- [AWS Systems Manager](#)
- [Visão geral de segurança do AWS Lambda](#)
- [Amazon CodeGuru](#)

- [Gerenciamento aprimorado e automatizado de vulnerabilidades para workloads de nuvem com um novo Amazon Inspector](#)
- [Automatizar o gerenciamento e a remediação de vulnerabilidades na AWS usando o Amazon Inspector e o AWS Systems Manager: Parte 1](#)

Vídeos relacionados:

- [Proteção de serviços com tecnologia sem servidor e de contêiner](#)
- [Práticas recomendadas de segurança para o serviço de metadados de instância do Amazon EC2](#)

SEC06-BP02 Provisionar computação por meio de imagens reforçadas

Ofereça menos oportunidades de acesso indesejado aos ambientes de runtime implantando-os com base em imagens reforçadas. Adquiria somente dependências de runtime, como imagens de contêiner e bibliotecas de aplicações, de registros confiáveis e verifique as respectivas assinaturas. Crie seus próprios registros privados para armazenar imagens e bibliotecas confiáveis para uso nos processos de criação e implantação.

Resultado desejado: seus recursos de computação são provisionados com base em imagens de referência reforçadas. Você recupera dependências externas, como imagens de contêiner e bibliotecas de aplicações, somente de registros confiáveis e verifica as respectivas assinaturas. Elas são armazenadas em registros privados para que seus processos de compilação e implantação as consultem. Você verifica e atualiza imagens e dependências regularmente para ajudar a oferecer proteção contra qualquer vulnerabilidade recém-descoberta.

Antipadrões comuns:

- Adquirir imagens e bibliotecas de registros confiáveis, mas não verificar a respectiva assinatura nem realizar verificações de vulnerabilidades antes de colocá-las em uso.
- Reforçar as imagens, mas não testá-las regularmente em busca de novas vulnerabilidades ou atualizá-las para a versão mais recente.
- Instalar ou não remover pacotes de software que não são necessários durante o ciclo de vida previsto da imagem.
- Confiar apenas na aplicação de patches para manter os recursos de computação de produção atualizados. Ao utilizar apenas a aplicação de patches, os recursos de computação podem se desviar do padrão reforçado com o passar do tempo. A aplicação de patches também pode não

conseguir remover malware que tenha sido instalado por um agente de ameaças durante um evento de segurança.

Benefícios do estabelecimento desta prática recomendada: o fortalecimento de imagens ajuda a reduzir o número de caminhos disponíveis em seu ambiente de runtime capazes de permitir acesso indesejado a usuários ou serviços não autorizados. Também pode reduzir o escopo do impacto caso ocorra algum acesso indesejado.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

Para fortalecer seus sistemas, comece com as versões mais recentes de sistemas operacionais, imagens de contêiner e bibliotecas de aplicações. Aplique patches aos problemas conhecidos. Minimize o sistema removendo quaisquer aplicações, serviços, drivers de dispositivo, usuários padrão e outras credenciais dispensáveis. Execute qualquer outra ação necessária, como desabilitar portas para criar um ambiente que tenha somente os recursos e capacidades essenciais para as workloads. Com base nesse parâmetro, você pode instalar software, agentes ou outros processos necessários para finalidades como monitoramento da workload ou gerenciamento de vulnerabilidades.

Você pode reduzir a sobrecarga de fortalecer os sistemas usando as orientações fornecidas por fontes confiáveis, como o [Center for Internet Security](#) (CIS) e os [Security Technical Implementation Guides \(STIGs\)](#) da Agência de Sistemas de Informação de Defesa (DISA). Recomendamos que você comece com uma [imagem de máquina da Amazon](#) (AMI) publicada pela AWS ou por um parceiro da APN e use o AWS [EC2 Image Builder](#) para automatizar a configuração de acordo com uma combinação apropriada de controles do CIS e do STIG.

Embora existam imagens reforçadas e fórmulas do EC2 Image Builder disponíveis que aplicam as recomendações do CIS ou do STIG da DISA, você pode descobrir que sua configuração impede que o software seja executado com êxito. Nessa situação, você pode começar com uma imagem base não reforçada, instalar o software e, em seguida, aplicar incrementalmente os controles do CIS para testar o respectivo impacto. Com relação a qualquer controle do CIS que impeça a execução do software, teste se é possível implementar as recomendações de fortalecimento mais refinadas em um STIG da DISA. Acompanhe os diferentes controles do CIS e as configurações do STIG da DISA que você pode aplicar com sucesso. Use-os para definir adequadamente suas fórmulas de fortalecimento de imagem no EC2 Image Builder.

Para workloads em contêineres, imagens reforçadas do Docker estão disponíveis no [repositório público](#) do [Amazon Elastic Container Registry \(ECR\)](#). Você pode usar o EC2 Image Builder para reforçar imagens de contêiner, bem como AMIs.

De modo semelhante aos sistemas operacionais e às imagens de contêiner, você pode obter pacotes de código (ou bibliotecas) de repositórios públicos por meio de determinadas ferramentas, como pip, npm, Maven e NuGet. Recomendamos que você gerencie pacotes de código integrando repositórios privados (por ex., dentro do [AWS CodeArtifact](#)) a repositórios públicos confiáveis. Com essa integração, você não precisa se preocupar em lidar com a recuperação, o armazenamento e a manutenção de pacotes atualizados. Seus processos de criação de aplicações podem então obter e testar a versão mais recente desses pacotes, bem como a aplicação, usando técnicas como análise de composição de software (SCA), testes estáticos de segurança de aplicações (SAST) e testes dinâmicos de segurança de aplicações (DAST).

Para workloads sem servidor que usam o AWS Lambda, simplifique o gerenciamento de dependências de pacotes usando [camadas do Lambda](#). Use camadas do Lambda para configurar um conjunto de dependências padrão que são compartilhadas em diferentes funções em um arquivo independente. Você pode criar e manter camadas por meio de seu próprio processo de criação, fornecendo um meio centralizado para manter as funções atualizadas.

Etapas da implementação

- Fortaleça o sistema operacional. Use imagens básicas de fontes confiáveis como base para criar AMIs reforçadas. Use o [EC2 Image Builder](#) para ajudar a personalizar o software instalado nas imagens.
- Fortaleça os recursos em contêiner. Configure recursos em contêiner para atender às práticas recomendadas de segurança. Ao usar contêineres, implemente a [verificação de imagens do ECR](#) no pipeline de compilação e regularmente no repositório de imagens para procurar CVEs nos contêineres.
- Ao utilizar a implementação sem servidor com o AWS Lambda, use [camadas do Lambda](#) para separar o código da função da aplicação e bibliotecas dependentes compartilhadas. Configure a [assinatura de código](#) para o Lambda a fim de garantir que somente código confiável seja executado nas funções do Lambda.

Recursos

Práticas recomendadas relacionadas:

- [OPS05-BP05 Executar o gerenciamento de patches](#)

Vídeos relacionados:

- [Deep dive into AWS Lambda security](#)

Exemplos relacionados:

- [Quickly build STIG-compliant AMI using EC2 Image Builder](#)
- [Building better container images](#)
- [Using Lambda layers to simplify your development process](#)
- [Develop & Deploy AWS Lambda Layers using Serverless Framework](#)
- [Building end-to-end AWS DevSecOps CI/CD pipeline with open source SCA, SAST and DAST tools](#)

SEC06-BP03 Reduzir o gerenciamento manual e o acesso interativo

Use a automação para realizar tarefas de implantação, configuração, manutenção e investigação sempre que possível. Considere usar o acesso manual aos recursos de computação em casos de procedimentos de emergência ou em ambientes seguros (sandbox) quando a automação não estiver disponível.

Resultado desejado: scripts programáticos e documentos de automação (runbooks) capturam ações autorizadas em seus recursos de computação. Esses runbooks são iniciados automaticamente, por meio de sistemas de detecção de alterações, ou manualmente, quando a avaliação humana é necessária. O acesso direto aos recursos de computação só é disponibilizado em situações de emergência quando a automação não está disponível. Todas as atividades manuais são registradas em log e incorporadas a um processo de análise para melhorar continuamente os recursos de automação.

Antipadrões comuns:

- Usar o acesso interativo a instâncias do Amazon EC2 com protocolos como SSH ou RDP.
- Manter logins de usuários individuais, como `/etc/passwd` ou usuários locais do Windows.
- Compartilhar uma senha ou chave privada para acessar uma instância entre vários usuários.
- Instalar software e criar ou atualizar manualmente arquivos de configuração.

- Atualizar ou corrigir manualmente o software.
- Fazer login em uma instância para solucionar problemas.

Benefícios do estabelecimento desta prática recomendada: a realização de ações com automação ajuda a reduzir o risco operacional de alterações indesejadas e configurações incorretas. A remoção do uso do Secure Shell (SSH) e do Remote Desktop Protocol (RDP) para acesso interativo reduz o escopo do acesso aos seus recursos de computação. Isso elimina um caminho comum para ações não autorizadas. Capturar suas tarefas de gerenciamento de recursos de computação em documentos de automação e scripts programáticos oferece um mecanismo para definir e auditar todo o escopo das atividades autorizadas em um nível de detalhe refinado.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Fazer login em uma instância é uma abordagem clássica para administração do sistema. Após a instalação do sistema operacional do servidor, os usuários normalmente fazem login manualmente para configurar o sistema e instalar o software desejado. Durante o ciclo de vida do servidor, os usuários podem fazer login para realizar atualizações de software, aplicar patches, alterar configurações e solucionar problemas.

No entanto, o acesso manual apresenta vários riscos. Ele exige um servidor que escute as solicitações, como um serviço SSH ou RDP, e possa fornecer um possível caminho para o acesso não autorizado. Também aumenta o risco de erro humano associado à execução de etapas manuais. Isso pode resultar em incidentes de workload, corrupção ou destruição de dados ou outros problemas de segurança. O acesso humano também exige proteções contra o compartilhamento de credenciais, criando uma sobrecarga adicional de gerenciamento.

Para mitigar esses riscos, você pode implementar uma solução de acesso remoto baseada em agente, como o [AWS Systems Manager](#). O AWS Systems Manager Agent (SSM Agent) inicia um canal criptografado e, portanto, não depende da escuta de solicitações iniciadas externamente. Considere configurar o SSM Agent para [estabelecer esse canal em um endpoint da VPC](#).

O Systems Manager oferece controle refinado sobre como você pode interagir com as instâncias gerenciadas. Você define as automações a serem executadas, quem pode executá-las e quando elas podem ser executadas. O Systems Manager pode aplicar patches, instalar software e fazer alterações na configuração sem acesso interativo à instância. O Systems Manager também pode fornecer acesso a um shell remoto e registrar em log cada comando invocado, bem como

a respectiva saída, durante a sessão para logs e ao [Amazon S3](#). O [AWS CloudTrail](#) registra invocações de APIs do Systems Manager para inspeção.

Etapas da implementação

1. [Instale o AWS Systems Manager Agent](#) (SSM Agent) nas instâncias do Amazon EC2. Verifique se o SSM Agent está incluso e foi iniciado automaticamente como parte da configuração básica da AMI.
2. Verifique se os perfis do IAM associados aos perfis de instância do EC2 incluem a [política gerenciada do IAM](#) `AmazonSSMManagedInstanceCore`.
3. Desabilite o SSH, o RDP e outros serviços de acesso remoto em execução nas instâncias. Você pode fazer isso executando scripts configurados na seção de dados do usuário dos seus modelos de lançamento ou criando AMIs personalizadas com ferramentas como o EC2 Image Builder.
4. Verifique se as regras de entrada do grupo de segurança aplicáveis às instâncias do EC2 não permitem acesso na porta 22/tcp (SSH) ou na porta 3389/tcp (RDP). Implemente a detecção e o alerta de grupos de segurança configurados incorretamente usando serviços como o AWS Config.
5. Defina automações, runbooks e comandos de execução apropriados no Systems Manager. Use políticas do IAM para definir quem pode realizar essas ações e as condições sob as quais elas são permitidas. Teste essas automações minuciosamente em um ambiente que não seja de produção. Invoque essas automações quando necessário, em vez de acessar a instância de forma interativa.
6. Use o [AWS Systems Manager Session Manager](#) para fornecer acesso interativo às instâncias quando necessário. Ative o registro em log de atividades da sessão para manter uma trilha de auditoria no [Amazon CloudWatch Logs](#) ou no [Amazon S3](#).

Recursos

Práticas recomendadas relacionadas:

- [REL08-BP04 Implantar usando infraestrutura imutável](#)

Exemplos relacionados:

- [Replacing SSH access to reduce management and security overhead with AWS Systems Manager](#)

Ferramentas relacionadas:

- [AWS Systems Manager](#)

Vídeos relacionados:

- [Controlling User Session Access to Instances in AWS Systems Manager Session Manager](#)

SEC06-BP04 Validar a integridade do software

Use a verificação criptográfica para validar a integridade dos artefatos de software (incluindo imagens) que a workload usa. Assine criptograficamente seu software como uma proteção contra alterações não autorizadas executadas em seus ambientes de computação.

Resultado desejado: todos os artefatos são obtidos de fontes confiáveis. Os certificados do site do fornecedor são validados. Os artefatos baixados são verificados criptograficamente com base na respectiva assinatura. Seu próprio software é assinado e verificado criptograficamente por seus ambientes de computação.

Antipadrões comuns:

- Confiar em sites de fornecedores confiáveis para obter artefatos de software, mas ignorar os avisos de expiração de certificado. Prosseguir com os downloads sem confirmar se os certificados são válidos.
- Validar certificados de sites de fornecedores, mas não verificar criptograficamente os artefatos baixados desses sites.
- Confiar apenas em resumos ou hashes para validar a integridade do software. Os hashes estabelecem que os artefatos não foram modificados da versão original, mas não validam a respectiva fonte.
- Não assinar seu próprio software, código ou biblioteca, mesmo quando usados apenas em suas próprias implantações.

Benefícios do estabelecimento desta prática recomendada: validar a integridade dos artefatos dos quais a workload depende ajuda a impedir a entrada de malware em seus ambientes de computação. Assinar seu software ajuda a impedir a execução não autorizada em seus ambientes de computação. Proteja sua cadeia de suprimentos de software assinando e verificando o código.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

As imagens do sistema operacional, as imagens de contêiner e os artefatos de código geralmente são distribuídos com verificações de integridade disponíveis, como por meio de um resumo ou hash. Isso permite que os clientes verifiquem a integridade calculando o hash da carga útil e validando se ele é o mesmo que o publicado. Embora essas verificações ajudem a conferir se a carga útil não foi adulterada, elas não validam se a carga útil veio da fonte original (sua procedência). A verificação de procedência exige um certificado emitido por uma autoridade confiável para assinar o artefato digitalmente.

Se você estiver usando um software ou artefatos baixados na workload, verifique se o provedor fornece uma chave pública para verificação de assinatura digital. Veja alguns exemplos de como a AWS fornece uma chave pública e instruções de verificação para o software que publicamos:

- [EC2 Image Builder: Verify the signature of the AWSTOE installation download](#)
- [AWS Systems Manager: Verifying the signature of SSM Agent](#)
- [Amazon CloudWatch: Verifying the signature of the CloudWatch agent package](#)

Incorpore a verificação de assinatura digital aos processos que você usa para obter e reforçar imagens, conforme discutido em [SEC06-BP02 Provisionar computação por meio de imagens reforçadas](#).

Você pode usar o [AWS Signer](#) para ajudar a gerenciar a verificação de assinaturas, bem como seu próprio ciclo de vida de assinatura de código para seus próprios programas de software e artefatos. Tanto o [AWS Lambda](#) quanto o [Amazon Elastic Container Registry](#) fornecem integrações com o Signer para verificar as assinaturas de código e imagens. Usando os exemplos na seção “Recursos”, você pode incorporar o Signer aos pipelines de integração e entrega contínuas (CI/CD) para automatizar a verificação de assinaturas e a assinatura de código e imagens.

Recursos

Documentos relacionados:

- [Cryptographic Signing for Containers](#)
- [Best Practices to help secure your container image build pipeline by using AWS Signer](#)
- [Announcing Container Image Signing with AWS Signer and Amazon EKS](#)
- [Configuring code signing for AWS Lambda](#)

- [Best practices and advanced patterns for Lambda code signing](#)
- [Code signing using AWS Certificate Manager Private CA and AWS Key Management Service asymmetric keys](#)

Exemplos relacionados:

- [Automate Lambda code signing with Amazon CodeCatalyst and AWS Signer](#)
- [Signing and Validating OCI Artifacts with AWS Signer](#)

Ferramentas relacionadas:

- [AWS Lambda](#)
- [AWS Signer](#)
- [AWS Certificate Manager](#)
- [AWS Key Management Service](#)
- [AWS CodeArtifact](#)

SEC06-BP05 Automatizar a proteção da computação

Automatize as operações de proteção da computação para reduzir a necessidade de intervenção humana. Use a verificação automatizada para detectar possíveis problemas em seus recursos de computação e corrigir com respostas programáticas automatizadas ou operações de gerenciamento de frota. Incorpore a automação em seus processos de CI/CD para implantar workloads confiáveis com dependências atualizadas.

Resultado desejado: os sistemas automatizados realizam toda a verificação e correção dos recursos de computação. Você usa a verificação automática para conferir se as imagens e dependências do software provêm de fontes confiáveis e se não foram adulteradas. As workloads são verificadas automaticamente quanto a dependências atualizadas e são assinadas para estabelecer confiabilidade em ambientes de computação da AWS. As correções automatizadas são iniciadas quando recursos irregulares são detectados.

Antipadrões comuns:

- Seguir a prática de infraestrutura imutável, mas sem ter uma solução para correção emergencial ou substituição de sistemas de produção.

- Usar a automação para corrigir recursos configurados incorretamente, mas sem ter um mecanismo de substituição manual instalado. Podem surgir situações em que você precise ajustar os requisitos e suspender as automações até fazer essas alterações.

Benefícios do estabelecimento desta prática recomendada: a automação pode reduzir o risco de acesso e uso não autorizados de seus recursos de computação. Isso ajuda a evitar que configurações incorretas entrem nos ambientes de produção e a detectar e corrigir configurações incorretas caso elas ocorram. A automação também ajuda a detectar acesso e uso não autorizados de recursos de computação para reduzir o tempo de resposta. Isso, por sua vez, pode reduzir o escopo geral do impacto do problema.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Você pode aplicar as automações descritas nas práticas do pilar de segurança para proteger seus recursos de computação. A prática [SEC06-BP01 Realizar o gerenciamento de vulnerabilidades](#) descreve como você pode usar o [Amazon Inspector](#) em seus pipelines de CI/CD e para verificar continuamente seus ambientes de runtime em busca de vulnerabilidades e exposições comuns (CVEs) conhecidas. Você pode usar o [AWS Systems Manager](#) para aplicar patches ou reimplantar imagens novas por meio de runbooks automatizados e manter sua frota de computação atualizada com os programas de software e as bibliotecas mais recentes. Use essas técnicas para reduzir a necessidade de processos manuais e acesso interativo aos seus recursos de computação. Consulte [SEC06-BP03 Reduzir o gerenciamento manual e o acesso interativo](#) para saber mais.

A automação também desempenha um papel na implantação de workloads confiáveis, descritas em [SEC06-BP02 Provisionar computação por meio de imagens reforçadas](#) e [SEC06-BP04 Validar a integridade do software](#). Você pode usar determinados serviços, como o [EC2 Image Builder](#), o [AWS Signer](#), o [AWS CodeArtifact](#) e o [Amazon Elastic Container Registry \(ECR\)](#), para baixar, verificar, construir e armazenar imagens e dependências de código reforçadas e aprovadas. Com o Inspector, cada um deles pode desempenhar um papel no processo de CI/CD, de forma que a workload chegue à produção somente quando for confirmado que suas dependências estão atualizadas e provêm de fontes confiáveis. A workload também é assinada para que os ambientes de computação da AWS, como [AWS Lambda](#) e [Amazon Elastic Kubernetes Service \(EKS\)](#), possam verificar se ela não foi adulterada antes de permitir sua execução.

Além desses controles preventivos, você também pode usar a automação nos controles de detecção para seus recursos de computação. A título de exemplo, o [AWS Security Hub](#) oferece o padrão

[NIST 800-53 Rev. 5](#), que inclui verificações como [\[EC2.8\] EC2 instances should use Instance Metadata Service Version 2 \(IMDSv2\)](#). O IMDSv2 usa as técnicas de autenticação de sessão, bloqueando solicitações que contêm um cabeçalho HTTP X-Forwarded-For e um TTL de rede de 1 para interromper o tráfego proveniente de fontes externas e recuperar informações sobre a instância do EC2. Essa verificação no Security Hub pode detectar quando as instâncias do EC2 usam o IMDSv1 e iniciar a correção automática. Saiba mais sobre detecção e correções automatizadas em [SEC04-BP04 Iniciar a correção de recursos irregulares](#).

Etapas da implementação

1. Automatize a criação de AMIs seguras, reforçadas e em conformidade com o [EC2 Image Builder](#). Você pode produzir imagens que incorporem controles dos padrões de referência do Center for Internet Security (CIS) ou do Security Technical Implementation Guide (STIG) com base em imagens básicas da AWS e de parceiros da APN.
2. Automatize o gerenciamento de configuração. Aplique e valide configurações seguras automaticamente em seus recursos de computação usando um serviço ou uma ferramenta de gerenciamento de configuração.
 - a. Gerenciamento automatizado de configuração usando o [AWS Config](#).
 - b. Gerenciamento automatizado de procedimento de segurança e conformidade usando o [AWS Security Hub](#).
3. Automatize a correção ou a substituição de instâncias do Amazon Elastic Compute Cloud (Amazon EC2). O Gerenciador de Patches do AWS Systems Manager automatiza o processo de aplicação de patches em instâncias gerenciadas com atualizações relacionadas à segurança e outros tipos de atualização. Você pode usar o Gerenciador de Patches para aplicar patches a sistemas operacionais e aplicações.
 - a. [Gerenciador de Patches do AWS Systems Manager](#)
4. Automatize a verificação de recursos de computação em busca de vulnerabilidades e exposições comuns (CVEs) e incorpore soluções de verificação de segurança em seu pipeline de criação.
 - a. [Amazon Inspector](#)
 - b. [Verificação de imagens do ECR](#)
5. Considere a possibilidade de usar o Amazon GuardDuty em detecção automática de malware e ameaças para proteger os recursos de computação. O GuardDuty também pode identificar possíveis problemas quando uma função do [AWS Lambda](#) é invocada em seu ambiente da AWS.
 - a. [Amazon GuardDuty](#)

6. Considere as soluções dos parceiros da AWS. Os parceiros da AWS oferecem produtos líderes do setor que são equivalentes, idênticos ou se integram aos controles existentes nos seus ambientes on-premises. Esses produtos complementam os serviços existentes da AWS para que você possa implantar uma arquitetura de segurança abrangente e obter uma experiência mais uniforme na nuvem e no ambiente on-premises.
 - a. [Segurança da infraestrutura](#)

Recursos

Práticas recomendadas relacionadas:

- [SEC01-BP06 Automatizar a implantação de controles de segurança padrão](#)

Documentos relacionados:

- [Get the full benefits of IMDSv2 and disable IMDSv1 across your AWS infrastructure](#)

Vídeos relacionados:

- [Security best practices for the Amazon EC2 instance metadata service](#)

Proteção de dados

Antes de criar a arquitetura de qualquer carga de trabalho, devem ser adotadas práticas fundamentais que influenciam a segurança. Por exemplo, a classificação de dados fornece uma maneira de categorizar os dados organizacionais com base nos níveis de sensibilidade, e a criptografia protege os dados tornando-os ininteligíveis ao acesso não autorizado. Esses métodos são importantes porque sustentam objetivos como evitar perdas financeiras ou cumprir obrigações regulatórias.

Na AWS, há várias abordagens a serem consideradas para lidar com a proteção de dados. A seção a seguir descreve como usar essas abordagens.

Tópicos

- [Classificação de dados](#)
- [Proteção de dados em repouso](#)
- [Proteção de dados em trânsito](#)

Classificação de dados

A classificação de dados fornece uma maneira de categorizar dados organizacionais com base em criticidade e confidencialidade para ajudá-lo a determinar os controles de proteção e retenção apropriados.

Práticas recomendadas

- [SEC07-BP01 Compreender seu esquema de classificação de dados](#)
- [SEC07-BP02 Aplicar controles de proteção de dados com base na confidencialidade dos dados](#)
- [SEC07-BP03 Automatizar a identificação e a classificação](#)
- [SEC07-BP04 Definir o gerenciamento escalável do ciclo de vida dos dados](#)

SEC07-BP01 Compreender seu esquema de classificação de dados

Compreenda a classificação dos dados que a workload está processando, os requisitos de tratamento, os processos de negócios associados, onde os dados são armazenados e quem é o proprietário dos dados. Seu esquema de classificação e tratamento de dados deve considerar

os requisitos legais e de conformidade aplicáveis à workload e quais controles de dados são necessários. Compreender os dados é a primeira etapa da jornada de classificação de dados.

Resultado desejado: os tipos de dados presentes na workload são bem compreendidos e documentados. Controles apropriados estão em vigor para proteger os dados confidenciais com base na respectiva classificação. Esses controles regem fatores como: quem tem permissão para acessar os dados e com que finalidade, onde eles são armazenados, a respectiva política de criptografia e como as chaves de criptografia são gerenciadas, o ciclo de vida dos dados e os requisitos de retenção, os processos de destruição apropriados, quais processos de backup e recuperação estão em vigor e a auditoria do acesso.

Antipadrões comuns:

- Não ter uma política formal de classificação de dados em vigor para definir os níveis de confidencialidade dos dados e os requisitos de tratamento.
- Não ter uma boa compreensão dos níveis de confidencialidade dos dados na workload e não capturar essas informações na documentação de arquitetura e operações.
- Não aplicar os controles apropriados sobre os dados com base na confidencialidade e nos respectivos requisitos, conforme descrito em sua política de classificação e tratamento de dados.
- Não fornecer feedback sobre os requisitos de classificação e tratamento de dados aos proprietários das políticas.

Benefícios do estabelecimento desta prática recomendada: esta prática elimina a ambiguidade com relação ao tratamento adequado dos dados na workload. A aplicação de uma política formal que defina os níveis de confidencialidade dos dados em sua organização e as proteções necessárias pode ajudar você a cumprir as regulamentações legais e outros atestados e certificações de segurança cibernética. Os proprietários das workloads podem ter certeza sobre onde os dados confidenciais estão armazenados e quais controles de proteção estão em vigor. Capturá-los na documentação ajuda os novos membros da equipe a compreendê-los melhor e a manter os controles no início de sua gestão. Essas práticas também podem ajudar a reduzir os custos ao dimensionar corretamente os controles para cada tipo de dados.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

Ao projetar uma workload, você pode considerar opções para proteger os dados confidenciais de forma intuitiva. Por exemplo, em uma aplicação multilocatário, é intuitivo considerar os dados de

cada locatário como confidenciais e implementar proteções para impedir que um locatário acesse os dados de outro locatário. Da mesma forma, você pode projetar controles de acesso intuitivamente para que apenas os administradores modifiquem os dados e os outros usuários tenham acesso somente leitura ou não tenham nenhum acesso.

Com esses níveis de confidencialidade de dados definidos e capturados na política, bem como os respectivos requisitos de proteção de dados, você pode identificar formalmente quais dados residem na workload. Em seguida, é possível determinar se os controles corretos estão em vigor, se os controles podem ser auditados e quais respostas são apropriadas se os dados forem tratados incorretamente.

Para ajudar a categorizar quando há dados confidenciais na workload, considere usar [tags de recursos](#) quando disponíveis. Por exemplo, você pode aplicar uma tag que tenha uma chave de tag de classificação e um valor de tag de PHI para informações de saúde protegidas (PHI) e outra tag que tenha uma chave de tag de confidencialidade e um valor de tag definido como alto. Serviços como o [AWS Config](#) podem então ser usados para monitorar esses recursos em busca de alterações e alertar se eles forem modificados de uma forma que anule a conformidade com seus requisitos de proteção (por ex., alterar as configurações de criptografia). Você pode capturar a definição padrão das chaves de tag e dos valores aceitáveis usando [políticas de tags](#), um recurso do AWS Organizations. Não é recomendável que a chave ou o valor da tag contenha dados privados ou confidenciais.

Etapas da implementação

1. Entenda o esquema de classificação de dados e os requisitos de proteção da sua organização.
2. Identifique os tipos de dados confidenciais processados pelas workloads.
3. Verifique se os dados confidenciais estão sendo armazenados e protegidos na workload de acordo com sua política. Use técnicas, como testes automatizados, para auditar a eficácia de seus controles.
4. Considere usar a marcação em nível de recursos e dados, quando disponível, para marcar o nível de confidencialidade dos dados e outros metadados operacionais que possam ajudar no monitoramento e resposta a incidentes.
 - a. As políticas de tag do AWS Organizations podem ser usadas para impor padrões de marcação.

Recursos

Práticas recomendadas relacionadas:

- [SUS04-BP01 Implementar uma política de classificação de dados](#)

Documentos relacionados:

- [Whitepaper Data Classification](#)
- [Best Practices for Tagging AWS Resources](#)

Exemplos relacionados:

- [AWS Organizations Tag Policy Syntax and Examples](#)

Ferramentas relacionadas

- [AWS Tag Editor](#)

SEC07-BP02 Aplicar controles de proteção de dados com base na confidencialidade dos dados

Aplice controles de proteção de dados que ofereçam um nível apropriado de controle para cada classe de dados definida em sua política de classificação. Essa prática pode permitir que você proteja dados confidenciais contra acesso e uso não autorizados, preservando a disponibilidade e o uso dos dados.

Resultado desejado: você tem uma política de classificação que define os diferentes níveis de confidencialidade dos dados em sua organização. Para cada um desses níveis de confidencialidade, você tem diretrizes claras publicadas para serviços e locais de armazenamento e manuseio aprovados e as respectivas configurações necessárias. Você implementa os controles para cada nível conforme o nível de proteção necessário e os custos correspondentes. Você dispõe de monitoramento e alertas para detectar se há dados em locais não autorizados, se eles são processados em ambientes não autorizados, se eles são acessados por agentes não autorizados ou se a configuração dos serviços relacionados está irregular.

Antipadrões comuns:

- Aplicar o mesmo nível de controles de proteção em todos os dados. Isso pode levar ao superprovisionamento de controles de segurança para dados com baixo nível de confidencialidade ou à proteção insuficiente dos dados altamente confidenciais.
- Não envolver as partes interessadas relevantes das equipes de segurança, conformidade e negócios ao definir os controles de proteção de dados.
- Ignorar as despesas operacionais indiretas e os custos associados à implementação e manutenção dos controles de proteção de dados.
- Não realizar revisões periódicas de controle de proteção de dados para manter o alinhamento com as políticas de classificação.

Benefícios do estabelecimento desta prática recomendada: ao alinhar seus controles com o nível de classificação de dados, sua organização pode investir em níveis mais altos de controle quando necessário. Isso pode incluir a ampliação dos recursos de proteção, monitoramento, medição, correção e geração de relatórios. Nas circunstâncias em que é apropriado usar menos controles, você pode melhorar a acessibilidade e a completude dos dados para seu quadro de funcionários, clientes ou membros. Essa abordagem possibilita que sua organização use os dados da forma mais flexível possível e, ao mesmo, cumpra os requisitos de proteção de dados.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

A implementação de controles de proteção de dados com base nos níveis de confidencialidade dos dados envolve várias etapas importantes. Primeiro, identifique os diferentes níveis de confidencialidade de dados em sua arquitetura de workload (como público, interno, confidencial e restrito) e avalie onde você armazena e processa esses dados. Em seguida, defina limites de isolamento em torno dos dados com base no respectivo nível de confidencialidade. Recomendamos que você separe os dados em diferentes Contas da AWS, usando [políticas de controle de serviços](#) (SCPs) a fim de restringir os serviços e ações permitidos para cada nível de confidencialidade de dados. Dessa forma, você pode criar limites de isolamento robustos e aplicar o princípio de privilégio mínimo.

Após a definição dos limites de isolamento, implemente controles de proteção apropriados com base nos níveis de confidencialidade dos dados. Consulte as práticas recomendadas para [proteção de dados em repouso](#) e [proteção de dados em trânsito](#) a fim de implementar controles relevantes, como criptografia, controles de acesso e auditoria. Considere técnicas como tokenização ou anonimização

para reduzir o nível de confidencialidade dos dados. Simplifique a aplicação de políticas de dados consistentes em sua empresa com um sistema centralizado para tokenização e destokenização.

Monitore e teste continuamente a eficácia dos controles implementados. Analise e atualize regularmente o esquema de classificação de dados, as avaliações de risco e os controles de proteção à medida que as ameaças e o cenário de dados de sua organização evoluírem. Alinhe os controles de proteção de dados implementados com as regulamentações, os padrões e os requisitos legais relevantes do setor. Além disso, ofereça conscientização e treinamento sobre segurança para ajudar os funcionários a entender o esquema de classificação de dados e suas responsabilidades no tratamento e proteção de dados confidenciais.

Etapas da implementação

1. Identifique os níveis de classificação e confidencialidade dos dados em sua workload.
2. Defina limites de isolamento para cada nível e determine uma estratégia de imposição.
3. Avalie os controles definidos por você que regem acesso, criptografia, auditoria, retenção e outras questões exigidas por sua política de classificação de dados.
4. Avalie as opções para reduzir o nível de confidencialidade dos dados quando apropriado, como usar tokenização ou anonimização.
5. Verifique os controles usando testes e monitoramento automatizados dos recursos configurados.

Recursos

Práticas recomendadas relacionadas:

- [PERF03-BP01 Usar um armazenamento de dados específico que melhor atenda aos seus requisitos de acesso e armazenamento de dados](#)
- [COST04-BP05 Reforçar políticas de retenção de dados](#)

Documentos relacionados:

- [Whitepaper Data Classification](#)
- [Best Practices for Security, Identify, & Compliance](#)
- [AWS KMS Best Practices](#)
- [Encryption best practices and features for AWS services](#)

Exemplos relacionados:

- [Building a serverless tokenization solution to mask sensitive data](#)
- [How to use tokenization to improve data security and reduce audit scope](#)

Ferramentas relacionadas:

- [AWS Key Management Service \(AWS KMS\)](#)
- [AWS CloudHSM](#)
- [AWS Organizations](#)

SEC07-BP03 Automatizar a identificação e a classificação

Automatizar a identificação e a classificação de dados pode ajudar a implementar os controles corretos. O uso da automação para ampliar a determinação manual reduz o risco de erro humano e a exposição.

Resultado desejado: é possível verificar se os controles adequados estão em vigor com base em sua política de classificação e tratamento. Ferramentas e serviços automatizados ajudam a identificar e classificar o nível de confidencialidade dos dados. A automação também ajuda a monitorar continuamente os ambientes para detectar e alertar se os dados estão sendo armazenados ou manipulados de forma não autorizada, para que medidas corretivas possam ser tomadas rapidamente.

Antipadrões comuns:

- Confiar apenas em processos manuais para identificação e classificação de dados, os quais podem ser propensos a erros e demorados. Isso pode resultar em uma classificação de dados ineficiente e inconsistente, especialmente à medida que os volumes de dados aumentam.
- Não ter mecanismos para monitorar e gerenciar ativos de dados em toda a organização.
- Ignorar a necessidade de monitoramento e classificação contínuos dos dados conforme eles se movimentam e evoluem dentro da organização.

Benefícios do estabelecimento desta prática recomendada: a automatização da identificação e da classificação de dados pode aumentar a consistência e a precisão da aplicação de controles de proteção de dados, reduzindo o risco de erro humano. A automação também pode fornecer

visibilidade sobre o acesso e a movimentação de dados confidenciais, ajudando a detectar o tratamento não autorizado e a tomar medidas corretivas.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Embora a avaliação humana seja frequentemente usada para classificar dados durante as fases iniciais de projeto de uma workload, considere a possibilidade de ter sistemas que automatizem a identificação e a classificação dos dados de teste como um controle preventivo. Por exemplo, os desenvolvedores podem receber uma ferramenta ou serviço para verificar dados representativos e determinar o nível de confidencialidade. Na AWS, você pode carregar conjuntos de dados no [Amazon S3](#) e verificá-los usando o [Amazon Macie](#), o [Amazon Comprehend](#) ou o [Amazon Comprehend Medical](#). Da mesma forma, considere a possibilidade de verificar os dados como parte dos testes de unidade e integração para detectar onde não deve haver dados confidenciais. Utilizar alertas sobre dados confidenciais nesse estágio pode destacar brechas nas proteções antes da implantação na produção. Outros recursos, como detecção de dados confidenciais no [AWS Glue](#), no [Amazon SNS](#) e no [Amazon CloudWatch](#), também podem ser usados para detectar PII e tomar medidas de mitigação. Com relação a quaisquer ferramentas ou serviços automatizados, entenda como eles definem dados confidenciais e complemente-os com outras soluções humanas ou automatizadas para resolver qualquer brecha conforme necessário.

Como controle de detecção, use o monitoramento contínuo dos ambientes para detectar se os dados confidenciais estão sendo armazenados de forma irregular. Isso pode ajudar a detectar determinadas situações, como publicação ou cópia de dados confidenciais em arquivos de log ou para um ambiente de análise de dados sem a devida desidentificação ou edição. Com relação aos dados armazenados no Amazon S3, os dados confidenciais podem ser monitorados continuamente usando o Amazon Macie.

Etapas da implementação

1. Execute uma verificação inicial dos ambientes para identificação e classificação automatizadas.
 - a. Uma verificação inicial completa dos dados pode ajudar a gerar uma compreensão abrangente do local em que os dados confidenciais residem nos ambientes. Quando uma verificação completa não for necessária inicialmente ou não puder ser concluída antecipadamente devido ao custo, avalie se as técnicas de amostragem de dados são adequadas para alcançar seus resultados. Por exemplo, o Amazon Macie pode ser configurado para realizar uma ampla operação automatizada de descoberta de dados confidenciais nos buckets do S3. Esse

recurso usa técnicas de amostragem para realizar de forma econômica uma análise preliminar do local em que os dados confidenciais residem. Uma análise mais detalhada dos buckets do S3 pode então ser realizada usando um trabalho de descoberta de dados confidenciais. Outros armazenamentos de dados também podem ser exportados para o S3 para serem verificados pelo Macie.

2. Configure verificações contínuas dos ambientes.
 - a. O recurso automatizado de descoberta de dados confidenciais do Macie pode ser usado para realizar verificações contínuas dos ambientes. Os buckets do S3 conhecidos e autorizados a armazenar dados confidenciais podem ser excluídos usando uma lista de permissões em no Macie.
3. Incorpore identificação e classificação nos processos de compilação e teste.
 - a. Identifique as ferramentas que os desenvolvedores podem usar para verificar a confidencialidade dos dados enquanto as workloads estão sendo desenvolvidas. Use essas ferramentas como parte dos testes de integração para emitir alertas quando houver dados confidenciais inesperados e evitar implantações adicionais.
4. Implemente um sistema ou um runbook para tomar medidas quando forem encontrados dados confidenciais em locais não autorizados.

Recursos

Documentos relacionados:

- [AWS Glue: Detect and process sensitive data](#)
- [Using managed data identifiers in Amazon SNS](#)
- [Amazon CloudWatch Logs: Help protect sensitive log data with masking](#)

Exemplos relacionados:

- [Enabling data classification for Amazon RDS database with Macie](#)
- [Detecting sensitive data in DynamoDB with Macie](#)

Ferramentas relacionadas:

- [Amazon Macie](#)
- [Amazon Comprehend](#)

- [Amazon Comprehend Medical](#)
- [AWS Glue](#)

SEC07-BP04 Definir o gerenciamento escalável do ciclo de vida dos dados

Entenda os requisitos do ciclo de vida dos dados relacionados aos seus diferentes níveis de classificação e tratamento de dados. Isso pode incluir como os dados são tratados quando entram pela primeira vez em seu ambiente, como eles são transformados e as regras de destruição. Considere fatores como períodos de retenção, acesso, auditoria e rastreamento de procedência.

Resultado desejado: os dados são classificados o mais próximo possível do ponto e hora de ingestão. Quando a classificação de dados exige mascaramento, tokenização ou outros processos que reduzam o nível de confidencialidade, você executa essas ações o mais próximo possível do ponto e hora de ingestão.

Você exclui os dados de acordo com sua política quando não é mais apropriado mantê-los e com base na respectiva classificação.

Antipadrões comuns:

- Implementar uma abordagem única de gerenciamento do ciclo de vida dos dados sem considerar os diferentes níveis de confidencialidade e requisitos de acesso.
- Considerar o gerenciamento do ciclo de vida somente do ponto de vista dos dados utilizáveis ou dos dados submetidos a backup, mas não de ambos.
- Supor que os dados que entraram na workload são válidos, sem estabelecer o respectivo valor ou procedência.
- Confiar na durabilidade dos dados como substituto dos backups e da proteção de dados.
- Reter os dados depois que eles já perderam a utilidade e após o período de retenção exigido.

Benefícios do estabelecimento desta prática recomendada: uma estratégia de gerenciamento do ciclo de vida dos dados bem definida e escalável ajuda a manter a conformidade regulatória, melhora a segurança dos dados, otimiza os custos de armazenamento e permite que os dados sejam acessados e compartilhados com eficiência, além de manter os controles apropriados.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

Os dados em uma workload geralmente são dinâmicos. A forma que eles assumem ao entrar no ambiente da workload pode ser diferente de quando são armazenados ou usados em lógica de negócios, relatórios, análises ou machine learning. Além disso, a importância dos dados pode mudar com o tempo. Alguns dados são de natureza temporal e perdem o valor à medida que se tornam obsoletos. Considere como essas mudanças nos dados afetam a avaliação em seu esquema de classificação de dados e controles associados. Sempre que possível, use um mecanismo de ciclo de vida automatizado, como [políticas de ciclo de vida do Amazon S3](#) e o [Amazon Data Lifecycle Manager](#), para configurar os processos de retenção, arquivamento e expiração de dados.

Diferencie os dados que estão disponíveis para uso e os dados armazenados como backup.

Considere usar o [AWS Backup](#) para automatizar o backup de dados nos serviços da AWS. Os [snapshots do Amazon EBS](#) oferecem uma forma de copiar um volume do EBS e armazená-lo por meio dos recursos do S3, incluindo ciclo de vida, proteção de dados e acesso a mecanismos de proteção. Dois desses mecanismos são o [Bloqueio de Objetos do S3](#) e a [Trava de Segurança do AWS Backup](#), que podem fornecer segurança e controle adicionais sobre seus backups. Gerencie a separação clara de deveres e acesso para backups. Isole os backups no nível da conta para manter a separação do ambiente afetado durante um evento.

Outro aspecto do gerenciamento do ciclo de vida é registrar o histórico dos dados à medida que eles progridem na workload, processo chamado de rastreamento da procedência dos dados. Desse modo, você pode ter certeza de que sabe de onde os dados vieram, quais transformações foram realizadas, qual proprietário ou processo fez essas alterações e quando. Ter esse histórico ajuda a solucionar problemas e investigações durante possíveis eventos de segurança. Por exemplo, os metadados sobre transformações podem ser registrados em log em uma tabela do [Amazon DynamoDB](#). Em um data lake, você pode manter cópias dos dados transformados em diferentes buckets do S3 para cada estágio do pipeline de dados. Armazene informações de esquema e carimbo de data e hora em um [AWS Glue Data Catalog](#). Independentemente de sua solução, considere os requisitos dos usuários finais para determinar as ferramentas apropriadas e necessárias para oferecer um relatório sobre a procedência dos dados. Isso ajudará você a determinar a melhor forma de rastrear a procedência.

Etapas da implementação

1. Analise os tipos de dados, os níveis de confidencialidade e os requisitos de acesso da workload para classificar os dados e definir estratégias apropriadas de gerenciamento do ciclo de vida.

2. Projete e implemente políticas de retenção de dados e processos automatizados de destruição que se alinhem aos requisitos legais, regulatórios e organizacionais.
3. Estabeleça processos e automação para monitoramento, auditoria e ajuste contínuos de estratégias, controles e políticas de gerenciamento do ciclo de vida dos dados à medida que os requisitos e as regulamentações da workload evoluem.

Recursos

Práticas recomendadas relacionadas:

- [COST04-BP05 Reforçar políticas de retenção de dados](#)
- [SUS04-BP03 Usar políticas para gerenciar o ciclo de vida de seus conjuntos de dados](#)

Documentos relacionados:

- [Whitepaper Data classification](#)
- [AWS Blueprint for Ransomware Defense](#)
- [DevOps Guidance: Improve traceability with data provenance tracking](#)

Exemplos relacionados:

- [How to protect sensitive data for its entire lifecycle in AWS](#)
- [Build data lineage for data lakes using AWS Glue, Amazon Neptune, and Spline](#)

Ferramentas relacionadas:

- [AWS Backup](#)
- [Amazon Data Lifecycle Manager](#)
- [AWS Identity and Access Management Access Analyzer](#)

Proteção de dados em repouso

Dados em repouso representam todos os dados mantidos no armazenamento não volátil por qualquer período na carga de trabalho. Isso inclui armazenamento em bloco, armazenamento de objetos, bancos de dados, arquivos, dispositivos IoT e qualquer outro meio de armazenamento no

qual os dados persistam. Proteger seus dados em repouso reduz o risco de acesso não autorizado quando a criptografia e os controles de acesso adequados são implementados.

Criptografia e tokenização são dois esquemas importantes mas distintos de proteção de dados.

Tokenização é um processo que permite definir um token para representar uma informação confidencial (por exemplo, o número do cartão de crédito de um cliente). Um token não deve ter um significado por si só nem deve ser derivado dos dados que ele tokeniza. Portanto, um resumo criptográfico não pode ser utilizado como um token. Ao definir cuidadosamente a abordagem de tokenização, você pode fornecer proteção adicional ao seu conteúdo e garantir o cumprimento dos requisitos de conformidade. Por exemplo, você pode reduzir o escopo de conformidade de um sistema de processamento de cartão de crédito se utilizar um token em vez de um número de cartão de crédito.

Criptografia é uma maneira de transformar o conteúdo de forma a torná-lo ilegível sem uma chave secreta para descriptografar o conteúdo novamente em texto sem formatação. Tanto a tokenização quanto a criptografia podem ser usadas para guardar e proteger as informações conforme apropriado. Além disso, o mascaramento é uma técnica que permite que parte dos dados seja editada até um ponto em que os dados restantes não são considerados confidenciais. Por exemplo, o PCI-DSS permite que os últimos quatro dígitos de um número de cartão sejam retidos fora do limite de escopo de conformidade para indexação.

Auditoria do uso de chaves de criptografia: entenda e faça a auditoria do uso de chaves de criptografia para confirmar se os mecanismos de controle de acesso nas chaves foram implementados adequadamente. Por exemplo, qualquer serviço da AWS que use uma chave do AWS KMS registra cada uso no AWS CloudTrail. Em seguida, você pode consultar o AWS CloudTrail usando uma ferramenta como o Amazon CloudWatch Insights para garantir que todos os usos de suas chaves sejam válidos.

Práticas recomendadas

- [SEC08-BP01 Implementar gerenciamento de chaves seguro](#)
- [SEC08-BP02 Aplicar criptografia em repouso](#)
- [SEC08-BP03 Automatizar a proteção de dados em repouso](#)
- [SEC08-BP04 Impor o controle de acesso](#)

SEC08-BP01 Implementar gerenciamento de chaves seguro

O gerenciamento seguro de chaves inclui o armazenamento, a rotação, o controle de acesso e o monitoramento do material essencial necessário para proteger os dados em repouso para sua workload.

Resultado desejado: um mecanismo de gerenciamento de chaves escalável, repetível e automatizado. O mecanismo deve fornecer a capacidade de impor o acesso de privilégio mínimo ao material essencial e fornecer o equilíbrio correto entre disponibilidade, confidencialidade e integridade das chaves. O acesso às chaves deve ser monitorado e o material da chave deve ser rotacionado por meio de um processo automatizado. O material de chave nunca deve estar acessível a identidades humanas.

Antipadrões comuns:

- Acesso humano a material de chave não criptografado.
- Criação de algoritmos criptográficos personalizados.
- Permissões excessivamente amplas para acessar materiais importantes.

Benefícios de estabelecer esta prática recomendada: Ao estabelecer um mecanismo seguro de gerenciamento de chaves para sua workload, você pode ajudar a proteger seu conteúdo contra acesso não autorizado. Além disso, você pode estar sujeito aos requisitos regulamentares para criptografar seus dados. Uma solução eficaz de gerenciamento de chaves pode fornecer mecanismos técnicos alinhados a essas regulamentações para proteger o material chave.

Nível de risco exposto se esta prática recomendada não for estabelecida: alto

Orientação para implementação

Muitos requisitos regulatórios e práticas recomendadas incluem a criptografia de dados em repouso como um controle de segurança fundamental. Para cumprir esse controle, sua workload precisa de um mecanismo para armazenar e gerenciar com segurança o material chave usado para criptografar seus dados em repouso.

A AWS oferece o AWS Key Management Service (AWS KMS) para fornecer armazenamento durável, seguro e redundante para chaves do AWS KMS. [Muitos serviços da AWS se integram com o AWS KMS](#) para oferecer suporte à criptografia de seus dados. O AWS KMS usa módulos de segurança de hardware validados pelo FIPS 140-2 Nível 3 para proteger suas chaves. Não há mecanismo para exportar chaves do AWS KMS em texto simples.

Ao implantar workloads usando uma estratégia de várias contas, isso é considerado [prática recomendada](#) para manter chaves do AWS KMS na mesma conta da workload que as usa. Nesse modelo distribuído, a responsabilidade pelo gerenciamento das chaves do AWS KMS é da equipe de aplicações. Em outros casos de uso, as organizações podem optar por armazenar as chaves do AWS KMS em uma conta centralizada. Essa estrutura centralizada requer políticas adicionais para permitir o acesso entre contas necessário para que a conta da workload acesse as chaves armazenadas na conta centralizada, mas pode ser mais aplicável em casos de uso em que uma única chave é compartilhada entre várias Contas da AWS.

Independentemente de onde o material da chave esteja armazenado, o acesso à chave deve ser rigorosamente controlado por meio do uso de [políticas de chave](#) e políticas do IAM. Políticas de chave são a principal forma de controlar o acesso a uma chave do AWS KMS. Além disso, concessões à chave do AWS KMS podem fornecer acesso a serviços da AWS para criptografar e descriptografar dados em seu nome. Reserve um tempo para revisar as [práticas recomendadas para controle de acesso às chaves do AWS KMS](#).

É uma prática recomendada monitorar o uso de chaves de criptografia para detectar padrões de acesso incomuns. As operações realizadas usando chaves gerenciadas pela AWS e chaves gerenciadas pelo cliente armazenadas no AWS KMS podem ser registradas no AWS CloudTrail e devem ser revisadas periodicamente. Atenção especial deve ser dada ao monitoramento dos principais eventos de destruição. Para mitigar a destruição acidental ou maliciosa de material de chave, os eventos de destruição da chave não excluem o material da chave imediatamente. As tentativas de excluir as chaves no AWS KMS estão sujeitas a um [período de espera](#), cujo padrão é de 30 dias, dando aos administradores tempo para revisar essas ações e reverter a solicitação, se necessário.

A maioria dos serviços da AWS usam o AWS KMS de forma transparente para você. Seu único requisito é decidir se quer usar uma chave gerenciada pela AWS ou gerenciada pelo cliente. Se sua workload exigir o uso direto de AWS KMS para criptografar ou descriptografar dados, a prática recomendada é usar [criptografia envelopada](#) para proteger seus dados. O [SDK de criptografia da AWS](#) pode fornecer às suas aplicações primitivas de criptografia do lado do cliente para implementar a criptografia envelopada e integrar com o AWS KMS.

Etapas da implementação

1. Determine as opções adequadas [de gerenciamento de chaves](#) (gerenciado pela AWS ou gerenciado pelo cliente) para a chave.

- Para facilitar o uso, a AWS oferece, para a maioria dos serviços, chaves de propriedade da AWS e gerenciadas por ela, que fornecem capacidade de criptografia em repouso sem a necessidade de gerenciar materiais ou políticas de chaves.
 - Ao usar chaves gerenciadas pelo cliente, considere o armazenamento de chaves padrão para fornecer o melhor equilíbrio entre agilidade, segurança, soberania de dados e disponibilidade. Outros casos de uso podem exigir o uso de armazenamentos de chaves personalizadas com [AWS CloudHSM](#) ou o [armazenamento de chaves externo](#).
2. Analise a lista de serviços que você está usando para sua workload para entender como o AWS KMS se integra ao serviço. Por exemplo, as instâncias do EC2 podem usar volumes criptografados do EBS, verificando se os snapshots do Amazon EBS criados a partir desses volumes também são criptografados usando uma chave gerenciada pelo cliente e mitigando a divulgação acidental de dados de snapshots não criptografados.
 - [Como os serviços da AWS são usados no AWS KMS](#)
 - Para obter informações detalhadas sobre as opções de criptografia que um serviço da AWS oferece, consulte o tópico Criptografia em repouso no guia do usuário ou no guia do desenvolvedor do serviço.
 3. Implemente o AWS KMS: o AWS KMS simplifica a criação e o gerenciamento de chaves e o controle do uso da criptografia em uma ampla variedade de serviços da AWS e em suas aplicações.
 - [Conceitos básicos: AWS Key Management Service \(AWS KMS\)](#)
 - Revise as [práticas recomendadas para controle de acesso às chaves do AWS KMS](#).
 4. Considere o AWS Encryption SDK: use a integração do AWS Encryption SDK com o AWS KMS quando sua aplicação precisar criptografar dados no lado do cliente.
 - [AWS Encryption SDK](#)
 5. Habilite o [IAM Access Analyzer](#) para revisar e notificar automaticamente se houver políticas de chave do AWS KMS excessivamente amplas.
 6. Habilite o [Security Hub](#) para receber notificações se houver políticas de chaves configuradas incorretamente, chaves programadas para exclusão ou chaves sem a rotação automática ativada.
 7. Determine o nível de registro em log apropriado para suas chaves do AWS KMS. Como as chamadas para o AWS KMS, incluindo eventos somente para leitura, são registradas em log, os logs do CloudTrail associados ao AWS KMS podem se tornar volumosos.
 - Algumas organizações preferem registrar a atividade de log do AWS KMS em uma trilha separada. Para obter mais detalhes, consulte a seção [Registro em log de chamadas de API do AWS KMS com CloudTrail](#) do guia do desenvolvedor do AWS KMS.

Recursos

Documentos relacionados:

- [AWS Key Management Service](#)
- [Ferramentas e serviços criptográficos da AWS](#)
- [Como proteger dados do Amazon S3 com o uso de criptografia](#)
- [Criptografia envelopada](#)
- [Promessa de soberania digital](#)
- [Desmistificação das operações de chave do AWS KMS, traga sua própria chave, armazenamento de chaves personalizado e portabilidade de texto cifrado](#)
- [Detalhes criptográficos do AWS Key Management Service](#)

Vídeos relacionados:

- [How Encryption Works in AWS \(Como funciona a criptografia na AWS\)](#)
- [Securing Your Block Storage on AWS \(Proteger seu armazenamento em bloco na AWS\)](#)
- [AWS data protection: Using locks, keys, signatures, and certificates \(Proteção de dados da AWS: uso de travas, chaves, assinaturas e certificados\)](#)

Exemplos relacionados:

- [Implemente mecanismos avançados de controle de acesso usando o AWS KMS](#)

SEC08-BP02 Aplicar criptografia em repouso

É necessário implementar o uso de criptografia para dados em repouso. A criptografia mantém a confidencialidade dos dados sigilosos em caso de acesso não autorizado ou divulgação acidental.

Resultado desejado: os dados privados devem ser criptografados por padrão quando em repouso. A criptografia ajuda a manter a confidencialidade dos dados e oferece uma camada adicional de proteção contra a divulgação intencional ou acidental de dados ou exfiltração. Dados criptografados não podem ser lidos nem acessados sem primeiro descriptografá-los. Todos os dados armazenados não criptografados devem ser inventariados e controlados.

Antipadrões comuns:

- Não utilizar configurações de criptografia por padrão.
- Conceder acesso excessivamente permissivo para chaves de descryptografia.
- Não monitorar o uso de chaves de criptografia e descryptografia.
- Armazenar dados não criptografados.
- Utilizar a mesma chave de criptografia para todos os dados, seja qual for o uso, os tipos e a classificação de dados.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientação de implementação

Mapeie as chaves de criptografia às classificações de dados em suas workloads. Essa abordagem ajuda a proteger-se contra o acesso excessivamente permissivo ao utilizar uma chave única ou um número muito pequeno de chaves de criptografia para seus dados (consulte [SEC07-BP01 Compreender seu esquema de classificação de dados](#)).

O AWS Key Management Service (AWS KMS) integra-se a muitos serviços da AWS para facilitar a criptografia de seus dados em repouso. Por exemplo, no Amazon Simple Storage Service (Amazon S3), você pode definir a [criptografia padrão](#) em um bucket para que os novos objetos sejam criptografados automaticamente. Ao utilizar o AWS KMS, considere o nível de restrição necessário para os dados. Chaves do AWS KMS controladas por serviço e padrão são gerenciadas e utilizadas em seu nome pelo AWS. Para dados sigilosos que exijam acesso refinado à chave de criptografia subjacente, considere chaves gerenciadas pelo cliente (CMKs). Você tem total controle sobre as CMKs, como gerenciamento de alternância e acesso pelo uso de políticas de chave.

Além disso, o [Amazon Elastic Compute Cloud \(Amazon EC2\)](#) e o [Amazon S3](#) são compatíveis com a imposição de criptografia ao definir a criptografia padrão. Você pode usar o [Regras do AWS Config](#) para conferir automaticamente se está usando criptografia, por exemplo, [volumes do Amazon Elastic Block Store \(Amazon EBS\)](#), instâncias do [Amazon Relational Database Service \(Amazon RDS\)](#) e [buckets do Amazon S3](#).

A AWS também oferece operações de criptografia do lado do cliente, possibilitando que você criptografe os dados antes de fazer upload deles para a nuvem. O AWS Encryption SDK oferece uma forma de criptografar seus dados utilizando a [criptografia envelopada](#). Você fornece a chave de encerramento, e o AWS Encryption SDK gera uma chave de dados exclusiva para cada objeto de dados que ele criptografa. Considere utilizar o AWS CloudHSM se precisar de um módulo de segurança de hardware de um locatário (HSM) gerenciado. O AWS CloudHSM possibilita gerar, importar e gerenciar chaves criptográficas em um HSM validado de nível 3 FIPS 140-2. Alguns

casos de uso do AWS CloudHSM incluem proteger chaves privadas para emitir uma autoridade de certificado (CA) e ativar a criptografia de dados transparente (TDE) para bancos de dados Oracle. O AWS CloudHSM Client SDK oferece software que possibilita criptografar dados do lado do cliente com chaves armazenadas no AWS CloudHSM antes de fazer upload de seus dados para AWS. O Amazon DynamoDB Encryption Client também possibilita criptografar e assinar itens antes de fazer upload para uma tabela do DynamoDB.

Etapas da implementação

- Impor criptografia em repouso para o Amazon S3: implemente a [criptografia padrão do bucket do Amazon S3](#).

Configurar a [criptografia padrão para volumes do Amazon EBS](#): especifique que você deseja que todos os volumes do Amazon EBS recém-criados sejam criados em formato criptografado, com a opção de usar a chave padrão fornecida pela AWS ou uma chave que você criar.

Configurar imagens de máquina da Amazon (AMIs) criptografadas: a cópia de uma AMI existente com criptografia habilitada criptografará automaticamente os volumes raiz e os snapshots.

Configurar a [criptografia do Amazon RDS](#): configure a criptografia para seus clusters de banco de dados Amazon RDS e snapshots em repouso utilizando a opção de criptografia.

Criar e configurar chaves do AWS KMS com políticas que limitem o acesso às entidades principais apropriadas para cada classificação de dados: por exemplo, crie uma chave do AWS KMS para criptografar dados de produção e uma chave diferente para criptografar dados de desenvolvimento ou teste. Você também pode conceder acesso de chave a outras Contas da AWS. Considere ter contas diferentes para seus ambientes de desenvolvimento e produção. Se seu ambiente de produção precisar descriptografar artefatos na conta de desenvolvimento, você poderá editar a política de CMK utilizada para criptografar os artefatos de desenvolvimento a fim de conferir à conta de produção a capacidade de descriptografar esses artefatos. O ambiente de produção pode, então, ingerir os dados descriptografados para uso na produção.

Configurar a criptografia em serviços da AWS adicionais: para outros serviços da AWS utilizados, leia a [documentação de segurança](#) do serviço em questão para determinar as opções de criptografia do serviço.

Recursos

Documentos relacionados:

- [Ferramentas de criptografia da AWS](#)
- [Documentação da AWS](#)
- [AWS Encryption SDK](#)
- [Whitepaper de detalhes criptográficos do AWS KMS](#)
- [AWS Key Management Service](#)
- [Ferramentas e serviços criptográficos da AWS](#)
- [Criptografia do Amazon EBS](#)
- [Criptografia padrão de volumes do Amazon EBS](#)
- [Criptografar recursos do Amazon RDS](#)
- [Como ativo a criptografia padrão para um bucket do Amazon S3?](#)
- [Proteger dados do Amazon S3 com o uso de criptografia](#)

Vídeos relacionados:

- [Como a criptografia funciona na AWS](#)
- [Proteger o armazenamento em bloco na AWS](#)

SEC08-BP03 Automatizar a proteção de dados em repouso

Use a automação para validar e aplicar controles de dados em repouso. Use a verificação automatizada para detectar configurações incorretas de soluções de armazenamento de dados e realize correções por meio de resposta programática automatizada sempre que possível. Incorpore a automação nos processos de CI/CD para detectar configurações incorretas de armazenamento de dados antes que elas sejam implantadas na produção.

Resultado desejado: os sistemas automatizados examinam e monitoram os locais de armazenamento de dados em busca de configurações incorretas de controles, acesso não autorizado e uso inesperado. A detecção de locais de armazenamento configurados incorretamente inicia correções automatizadas. Processos automatizados criam backups de dados e armazenam cópias imutáveis fora do ambiente original.

Antipadrões comuns:

- Não considerar as opções para habilitar a criptografia com configurações padrão, quando compatível.

- Não considerar eventos de segurança, além dos eventos operacionais, ao formular uma estratégia automatizada de backup e recuperação.
- Não impor configurações de acesso público para serviços de armazenamento.
- Não monitorar e auditar os controles para proteger os dados em repouso.

Benefícios do estabelecimento desta prática recomendada: a automação ajuda a evitar o risco de configurar incorretamente os locais de armazenamento de dados. Isso ajuda a evitar que configurações incorretas entrem nos ambientes de produção. Essa prática recomendada também ajuda a detectar e corrigir configurações incorretas, caso elas ocorram.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

A automação é um tema em todas as práticas para proteger os dados em repouso. A prática [SEC01-BP06 Automatizar a implantação de controles de segurança padrão](#) descreve como você pode capturar a configuração de seus recursos usando modelos de infraestrutura como código (IaC), como o [AWS CloudFormation](#). Esses modelos estão comprometidos com um sistema de controle de versão e são usados para implantar recursos da AWS por meio de um pipeline de CI/CD. Essas técnicas também se aplicam à automação da configuração de soluções de armazenamento de dados, como configurações de criptografia em buckets do Amazon S3.

Você pode verificar as configurações definidas nos modelos de IaC para verificar se há erros de configuração nos pipelines de CI/CD usando regras no [AWS CloudFormation Guard](#). Você pode monitorar configurações que ainda não estão disponíveis no CloudFormation ou em outras ferramentas de IaC para detectar erros de configuração com o [AWS Config](#). Os alertas gerados pelo Config para configurações incorretas podem ser corrigidos automaticamente, conforme descrito em [SEC04-BP04 Iniciar a correção de recursos irregulares](#).

Usar a automação como parte da estratégia de gerenciamento de permissões também é um componente essencial das proteções de dados automatizadas. As práticas [SEC03-BP02 Conceder acesso com privilégio mínimo](#) e [SEC03-BP04 Reduzir as permissões continuamente](#) descrevem a configuração de políticas de acesso de privilégio mínimo que são monitoradas continuamente pelo [AWS Identity and Access Management Access Analyzer](#) para gerar descobertas quando a permissão pode ser reduzida. Além da automação para monitorar permissões, você pode configurar o [Amazon GuardDuty](#) para observar comportamentos anômalos de acesso a dados nos [volumes do EBS](#) (por meio de uma instância do EC2), [buckets do S3](#) e [bancos de dados do Amazon Relational Database Service](#) compatíveis.

A automação também desempenha um papel para detectar o armazenamento de dados confidenciais em locais não autorizados. A prática [SEC07-BP03 Automatizar a identificação e a classificação](#) descreve como o [Amazon Macie](#) pode monitorar buckets do S3 em busca de dados confidenciais inesperados e gerar alertas que podem iniciar uma resposta automatizada.

Siga as práticas de [REL09 Back up data](#) para desenvolver uma estratégia automatizada de backup e recuperação de dados. O backup e a recuperação de dados são importantes para a recuperação tanto de eventos de segurança quanto de eventos operacionais.

Etapas da implementação

1. Capture a configuração de armazenamento de dados em modelos de IaC. Use verificações automatizadas nos pipelines de CI/CD para detectar configurações incorretas.
 - a. Você pode usá-las [<ulink type="marketing" url="cloudformation">&CFN;</ulink>](#) em modelos de IaC.
 - b. Use o [AWS Config](#) para executar regras em um modo de avaliação proativa. Use essa configuração como uma etapa em seu pipeline de CI/CD para verificar a conformidade de um recurso antes de criá-lo.
2. Monitore os recursos em busca de configurações incorretas de armazenamento de dados.
 - a. Defina o [AWS Config](#) para monitorar recursos de armazenamento de dados em busca de alterações nas configurações de controle e gerar alertas para invocar ações de correção ao detectar uma configuração incorreta.
 - b. Consulte [SEC04-BP04 Iniciar a correção de recursos irregulares](#) para obter mais orientações sobre correções automatizadas.
3. Monitore e reduza continuamente as permissões de acesso aos dados por meio da automação.
 - a. O [IAM Access Analyzer](#) pode ser executado continuamente para gerar alertas diante da possibilidade de as permissões serem reduzidas.
4. Monitore e emita alertas sobre comportamentos anômalos de acesso aos dados.
 - a. O [GuardDuty](#) supervisiona assinaturas de ameaças conhecidas e desvios dos comportamentos de acesso de referência aos recursos de armazenamento de dados, como volumes do EBS, buckets do S3 e bancos de dados do RDS.
5. Monitore e emita alertas sobre dados confidenciais armazenados em locais inesperados.
 - a. Use o [Amazon Macie](#) para verificar continuamente os buckets do S3 em busca de dados confidenciais.
6. Automatize backups seguros e criptografados dos dados.

- a. O [AWS Backup](#) é um serviço gerenciado que cria backups criptografados e seguros de várias fontes de dados na AWS. O [Elastic Disaster Recovery](#) permite que você copie workloads completas do servidor e mantenha a proteção contínua dos dados com um objetivo de ponto de recuperação (RPO) medido em segundos. Você pode configurar os dois serviços para que funcionem juntos e automatizem a criação de backups de dados e os copiem para locais de failover. Isso pode ajudar a manter os dados disponíveis quando eles forem afetados por eventos operacionais ou de segurança.

Recursos

Práticas recomendadas relacionadas:

- [SEC01-BP06 Automatizar a implantação de controles de segurança padrão](#)
- [SEC03-BP02 Conceder acesso com privilégio mínimo](#)
- [SEC03-BP04 Reduzir as permissões continuamente](#)
- [SEC04-BP04 Iniciar a correção de recursos irregulares](#)
- [SEC07-BP03 Automatizar a identificação e a classificação](#)
- [REL09-BP02 Proteger e criptografar backups](#)
- [REL09-BP03 Realizar o backup de dados automaticamente](#)

Documentos relacionados:

- [AWS Prescriptive Guidance: Automatically encrypt existing and new Amazon EBS volumes](#)
- [Ransomware Risk Management on AWS Using the NIST Cyber Security Framework \(CSF\)](#)

Exemplos relacionados:

- [How to use AWS Config proactive rules and AWS CloudFormation Hooks to prevent creation of noncompliant cloud resources](#)
- [Automate and centrally manage data protection for Amazon S3 with AWS Backup](#)
- [AWS re:Invent 2023 - Implement proactive data protection using Amazon EBS snapshots](#)
- [AWS re:Invent 2022 - Build and automate for resilience with modern data protection](#)

Ferramentas relacionadas:

- [AWS CloudFormation Guard](#)
- [AWS CloudFormation Guard Rules Registry](#)
- [IAM Access Analyzer](#)
- [Amazon Macie](#)
- [AWS Backup](#)
- [Elastic Disaster Recovery](#)

SEC08-BP04 Impor o controle de acesso

Para ajudar a proteger seus dados em repouso, implemente o controle de acesso utilizando mecanismos, como isolamento e versionamento, e aplique o princípio de privilégio mínimo. Evite conceder acesso público aos seus dados.

Resultado desejado: garantir que somente usuários autorizados possam acessar os dados conforme a necessidade. Proteger seus dados com backups regulares e versionamento a fim de impedir a modificação ou a exclusão de dados intencionais ou acidentais. Isolar dados críticos de outros dados a fim de proteger a confidencialidade e a integridade deles.

Antipadrões comuns:

- Armazenar dados com requisitos de confidencialidade ou classificações diferentes juntos.
- Utilizar permissões excessivamente permissivas em chaves de criptografia.
- Classificar dados de modo inadequado.
- Não reter backups detalhados de dados importantes.
- Conceder acesso persistente a dados de produção.
- Não auditar o acesso aos dados nem rever as permissões regularmente

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: baixo

Orientação de implementação

Vários controles podem ajudar a proteger seus dados em repouso, por exemplo, acesso (utilizando privilégio mínimo), isolamento e versionamento. Deve ser feita a auditoria de acesso aos seus dados com os mecanismos de detecção, como AWS CloudTrail e os logs de nível de serviço, como os logs

de acesso do Amazon Simple Storage Service (Amazon S3). Você deve inventariar quais dados são acessíveis publicamente e criar um plano para reduzir a quantidade de dados disponíveis ao longo do tempo.

O Amazon S3 Glacier Vault Lock e o Amazon S3 Object Lock fornecem controle de acesso obrigatório para os objetos no Amazon S3. Assim que uma política de cofre é bloqueada com a opção de conformidade, nem mesmo o usuário raiz pode alterá-la até que o bloqueio expire.

Etapas da implementação

- Aplicar o controle de acesso: aplique o controle de acesso com privilégios mínimos, incluindo acesso a chaves de criptografia.
- Dados separados com base em diferentes níveis de classificação: use diferentes Contas da AWS para níveis de classificação de dados e gerencie essas contas com o [AWS Organizations](#).
- Analisar as políticas do AWS Key Management Service (AWS KMS): [analise o nível de acesso](#) concedido nas políticas do AWS KMS.
- Revisar as permissões de objeto e de bucket do Amazon S3: revise regularmente o nível de acesso concedido nas políticas de bucket do S3. Uma das práticas recomendadas é evitar buckets que possam ser lidos ou gravados publicamente. Considere o uso do [AWS Config](#) para detectar buckets que estão disponíveis publicamente e do Amazon CloudFront para fornecer conteúdo do Amazon S3. Garanta que os buckets que não devem permitir acesso público sejam configurados adequadamente para evitar o acesso público. Por padrão, todos os buckets do S3 são privados e só ser acessados por usuários que receberam explicitamente esse acesso.
- Ativar o [AWS IAM Access Analyzer](#): o IAM Access Analyzer analisa os buckets do Amazon S3 e gera uma descoberta quando [uma política do S3 concede acesso a uma entidade externa](#).
- Habilitar o [versionamento do Amazon S3](#) e o [bloqueio de objetos](#) quando apropriado.
- Utilizar o [Amazon S3 Inventory](#): o Amazon S3 Inventory pode ser usado para auditar e gerar relatórios sobre o status de replicação e criptografia de seus objetos do S3.
- Revisar as permissões do [Amazon EBS](#) e do [compartilhamento de AMLs](#): as permissões de compartilhamento podem permitir que imagens e volumes sejam compartilhados com Contas da AWS externas à sua workload.
- Revise os compartilhamentos do [AWS Resource Access Manager](#) periodicamente para determinar se os recursos devem continuar a ser compartilhados. O Resource Access Manager possibilita compartilhar recursos, como políticas do AWS Network Firewall, regras do Amazon Route 53 Resolver e sub-redes em suas Amazon VPCs. Faça auditoria em recursos compartilhados regularmente e interrompa o compartilhamento dos que não precisem mais ser compartilhados.

Recursos

Práticas recomendadas relacionadas:

- [SEC03-BP01 Definir requisitos de acesso](#)
- [SEC03-BP02 Conceder acesso com privilégio mínimo](#)

Documentos relacionados:

- [Whitepaper de detalhes criptográficos do AWS KMS](#)
- [Introdução ao gerenciamento de permissões de acesso aos seus recursos do Amazon S3](#)
- [Visão geral do gerenciamento de acesso dos recursos do AWS KMS](#)
- [Regras do AWS Config](#)
- [Amazon S3 + Amazon CloudFront: uma combinação perfeita na nuvem](#)
- [Usar versionamento](#)
- [Bloquear objetos usando o Bloqueio de objetos do Amazon S3](#)
- [Compartilhar um snapshot do Amazon EBS](#)
- [AMIs compartilhadas](#)
- [Hospedar uma aplicação de uma página no Amazon S3](#)

Vídeos relacionados:

- [Proteger o armazenamento em bloco na AWS](#)

Proteção de dados em trânsito

Dados em trânsito são quaisquer dados enviados de um sistema para outro. Isso inclui a comunicação entre recursos em sua carga de trabalho, bem como a comunicação entre outros serviços e seus usuários finais. Ao fornecer o nível apropriado de proteção para os dados em trânsito, você protege a confidencialidade e a integridade dos dados de sua carga de trabalho.

Proteja dados entre VPC ou locais on-premises: Você pode usar o [AWS PrivateLink](#) para criar uma conexão de rede segura e privada entre Amazon Virtual Private Cloud (Amazon VPC) ou conectividade on-premises para serviços hospedados na AWS. Você pode acessar serviços da

AWS, serviços de terceiros e serviços em outras Contas da AWS como se estivessem em sua rede privada. Com o AWS PrivateLink, é possível acessar serviços em contas com CIDRs de IP sobrepostos sem precisar de um Gateway de Internet ou NAT. Você também não precisa configurar regras de firewall, definições de caminho ou tabelas de rotas. O tráfego permanece no backbone da Amazon e não atravessa a internet, portanto, seus dados estão protegidos. É possível manter a conformidade com os regulamentos de conformidade específicos do setor, como HIPAA e EU/US Privacy Shield. O AWS PrivateLink funciona perfeitamente com soluções de terceiros para criar uma rede global simplificada, permitindo acelerar a migração para a nuvem e aproveitar os serviços disponíveis da AWS.

Práticas recomendadas

- [SEC09-BP01 Implementar o gerenciamento seguro de chaves e certificados](#)
- [SEC09-BP02 Aplicar a criptografia em trânsito](#)
- [SEC09-BP03 Autenticar as comunicações de rede](#)

SEC09-BP01 Implementar o gerenciamento seguro de chaves e certificados

Os certificados Transport Layer Security (TLS) são usados para proteger as comunicações de rede e estabelecer a identidade de sites, recursos e workloads na internet, bem como em redes privadas.

Resultado desejado: Um sistema seguro de gerenciamento de certificados que pode provisionar, implantar, armazenar e renovar certificados em uma infraestrutura de chave pública (PKI). Um mecanismo seguro de gerenciamento de chaves e certificados evita que o material da chave privada do certificado seja divulgado e renova automaticamente o certificado periodicamente. Ele também se integra a outros serviços para fornecer comunicações de rede seguras e identidade para os recursos da máquina na workload. O material de chave nunca deve estar acessível a identidades humanas.

Antipadrões comuns:

- Executar etapas manuais durante os processos de implantação ou renovação de certificado.
- Não prestar a devida atenção à hierarquia da autoridade de certificação (CA) ao criar uma CA privada.
- Usar certificados autoassinados para recursos públicos.

Benefícios de estabelecer esta prática recomendada:

- Simplificar o gerenciamento de certificados por meio de implantação e renovação automatizadas.
- Incentivar a criptografia de dados em trânsito usando certificados TLS.
- Aumentar a segurança e a auditabilidade das ações de certificação realizadas pela autoridade de certificação.
- Organizar as tarefas de gerenciamento em diferentes camadas da hierarquia da CA.

Nível de risco exposto se esta prática recomendada não for estabelecida: alto

Orientação para implementação

As workloads modernas fazem uso extensivo de comunicações de rede criptografadas usando protocolos de PKI, como TLS. O gerenciamento de certificados PKI pode ser complexo, mas o provisionamento, a implantação e a renovação automatizados de certificados podem reduzir o atrito associado ao gerenciamento deles.

A AWS oferece dois serviços para gerenciar certificados de PKI de uso geral: [AWS Certificate Manager](#) e [AWS Private Certificate Authority \(AWS Private CA\)](#). O ACM é o principal serviço que os clientes usam para provisionar, gerenciar e implantar certificados para uso em workloads públicas e privadas da AWS. O ACM emite certificados usando o AWS Private CA e [integra-se](#) a muitos outros serviços gerenciados da AWS para fornecer certificados TLS seguros para workloads.

A AWS Private CA permite estabelecer a própria autoridade de certificação raiz ou subordinada e emitir certificados TLS por meio de uma API. É possível usar esses tipos de certificado em cenários em que você controla e gerencia a cadeia de confiança do lado do cliente da conexão TLS. Além dos casos de uso do TLS, a AWS Private CA pode ser usada para emitir certificados para pods do Kubernetes, atestados de produtos de dispositivos Matter, assinatura de código e outros casos de uso com um [modelo personalizado](#). Você também pode usar [IAM Roles Anywhere](#) para fornecer credenciais do IAM temporárias para workloads on-premises que recebem certificados X.509 assinados pela CA privada.

Além do ACM e do AWS Private CA, o [AWS IoT Core](#) oferece suporte especializado para provisionar, gerenciar e implantar certificados de PKI em dispositivos de IoT. O AWS IoT Core fornece mecanismos especializados para [integração de dispositivos de IoT](#) à infraestrutura de chave pública em grande escala.

Considerações para estabelecer uma hierarquia de CA privada

Quando precisar estabelecer uma CA privada, é importante tomar cuidado especial para projetar adequadamente a hierarquia da CA com antecedência. É uma prática recomendada implantar cada

nível de sua hierarquia de CA em Contas da AWS separadas ao criar uma hierarquia de CA privada. Essa etapa intencional reduz a área de superfície de cada nível na hierarquia da CA, simplificando a descoberta de anomalias nos dados de log do CloudTrail e reduzindo o escopo de acesso ou impacto se houver acesso não autorizado a uma das contas. A CA raiz deve residir em uma própria conta separada e deve ser usada somente para emitir um ou mais certificados de CA intermediários.

Depois, crie uma ou mais CAs intermediárias em contas separadas da conta da CA raiz para emitir certificados para usuários finais, dispositivos ou outras workloads. Por fim, emita certificados da CA raiz para as CAs intermediárias, que, por sua vez, emitirão certificados para os usuários finais ou dispositivos. Para obter mais informações sobre como planejar a implantação de CA e projetar a hierarquia de CA, incluindo planejamento de resiliência, replicação entre regiões, compartilhamento de CAs na organização e muito mais, consulte [Planning your AWS Private CA deployment](#).

Etapas da implementação

1. Determine os serviços da AWS relevantes e necessários para seu caso de uso:

- Muitos casos de uso podem aproveitar a infraestrutura de chave pública da AWS existente usando o [AWS Certificate Manager](#). O ACM pode ser usado para implantar certificados TLS para servidores web, balanceadores de carga ou outros usos para certificados publicamente confiáveis.
- Considere [AWS Private CA](#) quando precisar estabelecer a própria hierarquia de autoridade de certificação privada ou precisar acessar certificados exportáveis. O ACM pode então ser usado para emitir [muitos tipos de certificados de entidade final](#) usando a AWS Private CA.
- Para casos de uso em que os certificados devem ser provisionados em grande escala para dispositivos incorporados de Internet das Coisas (IoT), pense no [AWS IoT Core](#).

2. Implemente a renovação automática do certificado sempre que possível:

- Use [a renovação gerenciada pelo ACM](#) para certificados emitidos pelo ACM junto com serviços gerenciados da AWS integrados.

3. Estabeleça trilhas de auditoria e registro:

- Habilite o [Logs do CloudTrail](#) para monitorar o acesso às contas que têm autoridades de certificação. Considere configurar a validação da integridade do arquivo de log no CloudTrail para verificar a autenticidade dos dados de log.
- Gere e revise periodicamente [relatórios de auditoria](#) que listam os certificados que a CA privada emitiu ou revogou. Esses relatórios podem ser exportados para um bucket do S3.
- Ao implantar uma CA privada, você também precisará estabelecer um bucket do S3 para armazenar a lista de revogação de certificados (CRL). Para obter orientação sobre como

configurar esse bucket do S3 com base nos requisitos da workload, consulte [Planejar uma lista de revogação de certificados \(CRL\)](#).

Recursos

Práticas recomendadas relacionadas:

- [SEC02-BP02 Usar credenciais temporárias](#)
- [SEC08-BP01 Implementar gerenciamento de chaves seguro](#)
- [SEC09-BP03 Autenticar as comunicações de rede](#)

Documentos relacionados:

- [How to host and manage an entire private certificate infrastructure in AWS](#)
- [How to secure an enterprise scale ACM Private CA hierarchy for automotive and manufacturing](#)
- [Práticas recomendadas de CA privada](#)
- [How to use AWS RAM to share your ACM Private CA cross-account](#)

Vídeos relacionados:

- [Activating AWS Certificate Manager Private CA \(workshop\)](#)

Exemplos relacionados:

- [Workshop de CA privada](#)
- [IOT Device Management Workshop](#) (incluindo provisionamento de dispositivos)

Ferramentas relacionadas:

- [Plug-in para o gerenciador de certificados do Kubernetes para usar a AWS Private CA](#)

SEC09-BP02 Aplicar a criptografia em trânsito

Aplice os requisitos de criptografia definidos com base em políticas, obrigações regulatórias e padrões da organização para cumprir os requisitos organizacionais, legais e de conformidade. Utilize

somente protocolos com criptografia ao transmitir dados sigilosos para fora da sua nuvem privada virtual (VPC). A criptografia ajuda a manter a confidencialidade dos dados mesmo quando os dados passam por redes não confiáveis.

Resultado desejado: todos os dados devem ser criptografados em trânsito com pacotes de criptografia e protocolos TLS seguros. O tráfego de rede entre seus recursos e a Internet deve ser criptografado para reduzir o acesso não autorizado aos dados. O tráfego de rede exclusivamente em seu ambiente interno da AWS deve ser criptografado com TLS sempre que possível. A rede interna da AWS é criptografada por padrão e o tráfego de rede em uma VPC não pode ser adulterado nem interceptado a menos que uma parte não autorizada tenha obtido acesso ao recurso que esteja gerando o tráfego (como instâncias do Amazon EC2 e contêineres do Amazon ECS). Considere proteger o tráfego de rede para rede com uma rede privada virtual (VPN) IPsec.

Antipadrões comuns:

- Utilizar versões obsoletas de SSL, TLS e componentes do pacote de criptografia (por exemplo, SSL v3.0, chaves RSA de 1024 bits e criptografia RC4).
- Permitir tráfego não criptografado (HTTP) para ou de recursos voltados para o público.
- Não monitorar e substituir certificados X.509 antes da validade.
- Utilizar certificados X.509 autoassinados para TLS.

Nível de exposição a riscos quando esta prática recomendada não é estabelecida: alto

Orientação de implementação

Os serviços da AWS fornecem endpoints HTTPS usando TLS para comunicação, fornecendo criptografia em trânsito quando se comunicam com as APIs da AWS. Protocolos não seguros, como HTTP, podem ser auditados e bloqueados em uma VPC por meio do uso de grupos de segurança. Solicitações HTTP também podem ser [redirecionadas automaticamente para HTTPS](#) no Amazon CloudFront ou em um [Application Load Balancer](#). Você tem controle total sobre seus recursos de computação para implementar a criptografia em trânsito em seus serviços. Além disso, você pode usar a conectividade VPN em sua VPC a partir de uma rede externa ou [AWS Direct Connect](#) para facilitar a criptografia do tráfego. Verifique se os seus clientes estão fazendo chamadas para APIs da AWS utilizando pelo menos TLS 1.2, pois a [AWS tornará obsoleto o uso de TLS 1.0 e 1.1 em junho de 2023](#). Soluções de terceiros estão disponíveis no AWS Marketplace, caso você tenha requisitos especiais.

Etapas da implementação

- Aplicar a criptografia em trânsito: os requisitos de criptografia definidos devem se basear nos mais recentes padrões e práticas recomendadas e permitir apenas protocolos seguros. Por exemplo, configure apenas um grupo de segurança para permitir o protocolo HTTPS a um Application Load Balancer ou instância do Amazon EC2.
- Configurar protocolos seguros em serviços de borda: [configure o HTTPS com Amazon CloudFront](#) e utilize um [perfil de segurança apropriado para seu procedimento de segurança e caso de uso](#).
- Utilizar uma [VPN para conectividade externa](#): considere usar uma VPN IPsec para proteger conexões ponto a ponto ou rede a rede para fornecer privacidade e integridade dos dados.
- Configurar protocolos seguros em balanceadores de carga: selecione uma política de segurança que ofereça os pacotes de criptografia mais fortes compatíveis com os clientes que se conectarão ao receptor. [Criar um receptor de HTTPS para seu Application Load Balancer](#).
- Configurar protocolos seguros no Amazon Redshift: configure o cluster para exigir uma [conexão Secure Socket Layer \(SSL\) ou Transport Layer Security \(TLS\)](#).
- Configurar protocolos seguros: leia a documentação do serviço da AWS para determinar os recursos de criptografia em trânsito.
- Configurar o acesso seguro ao fazer upload para buckets do Amazon S3: utilize controles de política de bucket do Amazon S3 para [implementar acesso seguro](#) aos dados.
- Considerar o uso do [AWS Certificate Manager](#): o ACM permite fornecer, gerenciar e implantar certificados TLS públicos para uso com serviços da AWS.
- Considerar o uso do [AWS Private Certificate Authority](#) para necessidades de PKI privada: o AWS Private CA permite criar hierarquias de autoridade de certificado privada (CA) para emitir certificados X.509 entidade final que podem ser usados para criar canais de TLS criptografados.

Recursos

Documentos relacionados:

- [Documentação da AWS](#)
- [Utilizar HTTPS com o CloudFront](#)
- [Conectar sua VPC a redes remotas usando a AWS Virtual Private Network](#)
- [Criar um receptor de HTTPS para seu Application Load Balancer](#)
- [Tutorial: configurar o SSL/TLS no Amazon Linux 2](#)
- [Usar SSL/TLS para criptografar uma conexão com uma instância de banco de dados](#)
- [Configurar as opções de segurança para conexões](#)

SEC09-BP03 Autenticar as comunicações de rede

Verifique a identidade das comunicações usando protocolos que oferecem suporte à autenticação, como Transport Layer Security (TLS) ou IPsec.

Projete a workload para usar protocolos de rede seguros e autenticados sempre que for feita uma comunicação entre serviços, aplicações ou usuários. O uso de protocolos de rede compatíveis com a autenticação e a autorização fornece maior controle sobre os fluxos de rede e reduz o impacto do acesso não autorizado.

Resultado desejado: uma workload com fluxos de tráfego bem definidos do plano de dados e do ambiente de gerenciamento entre os serviços. Os fluxos de tráfego usam protocolos de rede autenticados e criptografados quando tecnicamente viáveis.

Antipadrões comuns:

- Fluxos de tráfego não criptografados ou não autenticados na workload.
- Reutilizar credenciais de autenticação entre vários usuários ou entidades.
- Confiar apenas nos controles de rede como um mecanismo de controle de acesso.
- Criar um mecanismo de autenticação personalizado em vez de depender de mecanismos de autenticação padrão do setor.
- Fluxos de tráfego excessivamente permissivos entre componentes de serviço ou outros recursos na VPC.

Benefícios do estabelecimento desta prática recomendada:

- Limita o escopo do impacto do acesso não autorizado a uma parte da workload.
- Fornece um nível mais alto de garantia de que as ações são executadas somente por entidades autenticadas.
- Melhora o desacoplamento de serviços definindo e aplicando claramente as interfaces de transferência de dados pretendidas.
- Melhora o monitoramento, o log e a resposta a incidentes por meio da atribuição de solicitações e interfaces de comunicação bem definidas.
- Oferece defesa profunda para as workloads combinando controles de rede com controles de autenticação e de autorização.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: baixo

Orientações para a implementação

Os padrões de tráfego de rede da workload podem ser caracterizados em duas categorias:

- O tráfego leste-oeste representa fluxos de tráfego entre serviços que compõem uma workload.
- O tráfego norte-sul representa fluxos de tráfego entre a workload e os consumidores.

Embora seja uma prática comum criptografar o tráfego norte-sul, é menos comum proteger o tráfego leste-oeste usando protocolos autenticados. As práticas modernas de segurança recomendam que o design da rede por si só não conceda um relacionamento confiável entre duas entidades. Quando dois serviços puderem residir dentro de um limite de rede comum, criptografar, autenticar e autorizar as comunicações ainda são práticas recomendadas entre esses serviços.

Como exemplo, as APIs de serviços da AWS usam o protocolo de assinatura do [Signature Version 4 \(SigV4\) da AWS](#) para autenticar o chamador, independentemente da rede de origem da solicitação. Essa autenticação garante que as APIs da AWS possam verificar a identidade que solicitou a ação e que essa identidade possa ser combinada com políticas para tomar uma decisão de autorização a fim de determinar se a ação deve ser permitida ou não.

Serviços, como o [Amazon VPC Lattice](#) e o [Amazon API Gateway](#) permitem usar o mesmo protocolo de assinatura SigV4 para adicionar autenticação e autorização ao tráfego leste-oeste em suas próprias workloads. Se os recursos fora do ambiente da AWS precisarem se comunicar com os serviços que exigem autenticação e autorização baseadas em SigV4, você poderá usar o [AWS Identity and Access Management \(IAM\) Roles Anywhere](#) no recurso que não é da AWS para adquirir credenciais temporárias da AWS. Essas credenciais podem ser usadas para assinar solicitações para serviços que usam o SigV4 para autorizar o acesso.

Outro mecanismo comum para autenticar o tráfego leste-oeste é a autenticação mútua TLS (mTLS). Muitas aplicações da Internet das Coisas (IoT), aplicações business to business e microsserviços usam o mTLS para validar a identidade de ambos os lados de uma comunicação TLS por meio do uso de certificados X.509 do lado do cliente e do servidor. Esses certificados podem ser emitidos por AWS Private Certificate Authority (AWS Private CA). É possível usar serviços como o [Amazon API Gateway](#) e o [AWS App Mesh](#) para fornecer autenticação mTLS para comunicação entre workloads ou dentro da workload. Embora o mTLS forneça informações de autenticação aos dois lados de uma comunicação TLS, ele não fornece um mecanismo de autorização.

Por fim, o OAuth 2.0 e o OpenID Connect (OIDC) são dois protocolos normalmente usados para controlar o acesso aos serviços pelos usuários, mas agora também estão se tornando populares para o tráfego entre serviços. O API Gateway fornece um [autorizador JSON Web Token \(JWT\)](#), que permite que as workloads restrinjam o acesso às rotas de API usando JWTs emitidos por provedores de identidades OIDC ou OAuth 2.0. Os escopos do OAuth2 podem ser usados como uma fonte para decisões básicas de autorização, mas as verificações de autorização ainda precisam ser implementadas na camada da aplicação, e os escopos do OAuth2 por si só não atendem a necessidades de autorização mais complexas.

Etapas da implementação

- Definir e documentar os fluxos de rede da workload: a primeira etapa na implementação de uma estratégia de defesa profunda é definir os fluxos de tráfego da workload.
 - Crie um diagrama de fluxo de dados que defina claramente como os dados são transmitidos entre os diferentes serviços que compõem a workload. Esse diagrama é a primeira etapa para aplicar esses fluxos por meio de canais de rede autenticados.
 - Instrumente a workload nas fases de desenvolvimento e testes para validar se o diagrama de fluxo de dados reflete com precisão o comportamento da workload em runtime.
 - Um diagrama de fluxo de dados também pode ser útil ao realizar um exercício de modelagem de ameaças, conforme descrito em [SEC01-BP07 Identificar ameaças e priorizar mitigações com o uso de um modelo de ameaça](#).
- Estabeleça controles de rede: considere os recursos da AWS para estabelecer controles de rede alinhados aos fluxos de dados. Embora os limites da rede não devam ser o único controle de segurança, eles fornecem uma camada na estratégia de defesa profunda para proteger a workload.
 - Use [grupos de segurança](#) para estabelecer, definir e restringir fluxos de dados entre recursos.
 - Considere usar o [AWS PrivateLink](#) para se comunicar com os serviços da AWS e de terceiros que são compatíveis com o AWS PrivateLink. Os dados enviados por meio de um endpoint da interface do AWS PrivateLink permanecem na estrutura da rede da AWS e não atravessam a internet pública.
- Implementar autenticação e autorização entre os serviços na workload: escolha o conjunto de serviços da AWS mais apropriado para fornecer fluxos de tráfego autenticados e criptografados na workload.
 - Considere o [Amazon VPC Lattice](#) para proteger a comunicação entre serviços. O VPC Lattice pode usar a [autenticação do SigV4 combinada com políticas de autenticação](#) para controlar o acesso entre serviços.

- Para comunicação entre serviços usando mTLS, considere o [API Gateway](#) ou o [App Mesh](#). O [AWS Private CA](#) pode ser usado para estabelecer uma hierarquia de CA privada capaz de emitir certificados para uso com o mTLS.
- Ao fazer a integração com serviços que usam OAuth 2.0 ou OIDC, considere o [API Gateway usando o autorizador JWT](#).
- Para comunicação entre a workload e dispositivos de IoT, considere o [AWS IoT Core](#), que fornece várias opções para criptografia e autenticação de tráfego de rede.
- Monitorar o acesso não autorizado: monitore continuamente os canais de comunicação indesejados, entidades principais não autorizadas que tentam acessar recursos protegidos e outros padrões de acesso inadequados.
 - Se estiver usando o VPC Lattice para gerenciar o acesso aos serviços, considere ativar e monitorar os [logs de acesso do VPC Lattice](#). Esses logs de acesso incluem informações sobre a entidade solicitante, informações de rede que incluem a VPC de origem e de destino e os metadados da solicitação.
 - Considere a ativação dos [Logs de fluxo da VPC](#) para capturar metadados nos fluxos de rede e analisar se há anomalias periodicamente.
 - Consulte o [AWS Security Incident Response Guide](#) e a seção [Resposta a incidentes](#) do Pilar Segurança: AWS Well-Architected Framework para obter mais orientações sobre planejamento, simulação e resposta a incidentes de segurança.

Recursos

Práticas recomendadas relacionadas:

- [SEC03-BP07 Analisar o acesso público e entre contas](#)
- [SEC02-BP02 Usar credenciais temporárias](#)
- [SEC01-BP07 Identificar ameaças e priorizar mitigações com o uso de um modelo de ameaça](#)

Documentos relacionados:

- [Evaluating access control methods to secure Amazon API Gateway APIs](#)
- [Configurar a autenticação TLS mútua para uma API REST](#)
- [How to secure API Gateway HTTP endpoints with JWT authorizer](#)
- [Authorizing direct calls to AWS services using AWS IoT Core credential provider](#)

- [AWS Security Incident Response Guide](#)

Vídeos relacionados:

- [AWS re:invent 2022: Introducing VPC Lattice](#)
- [AWS re:invent 2020: Serverless API authentication for HTTP APIs on AWS](#)

Exemplos relacionados:

- [Workshop do Amazon VPC Lattice](#)
- [Workshop Zero-Trust Episode 1 – The Phantom Service Perimeter](#)

Resposta a incidentes

Mesmo com controles preventivos e de detecção consolidados, sua organização deve implementar mecanismos para responder e atenuar o impacto potencial de incidentes de segurança.

Sua preparação afeta muito a capacidade das equipes de operar com eficácia durante um incidente, isolar, conter e analisar problemas e restaurar as operações para um estado adequado conhecido. Implementar as ferramentas e o acesso antes de um incidente de segurança e praticar rotineiramente dias de jogos para validar a resposta a incidentes ajudam a garantir que você possa se recuperar enquanto minimiza interrupções empresariais.

Tópicos

- [Aspectos da resposta a incidentes da AWS](#)
- [Elaborar objetivos da resposta da nuvem](#)
- [Preparação](#)
- [Operações](#)
- [Atividade pós-incidente](#)

Aspectos da resposta a incidentes da AWS

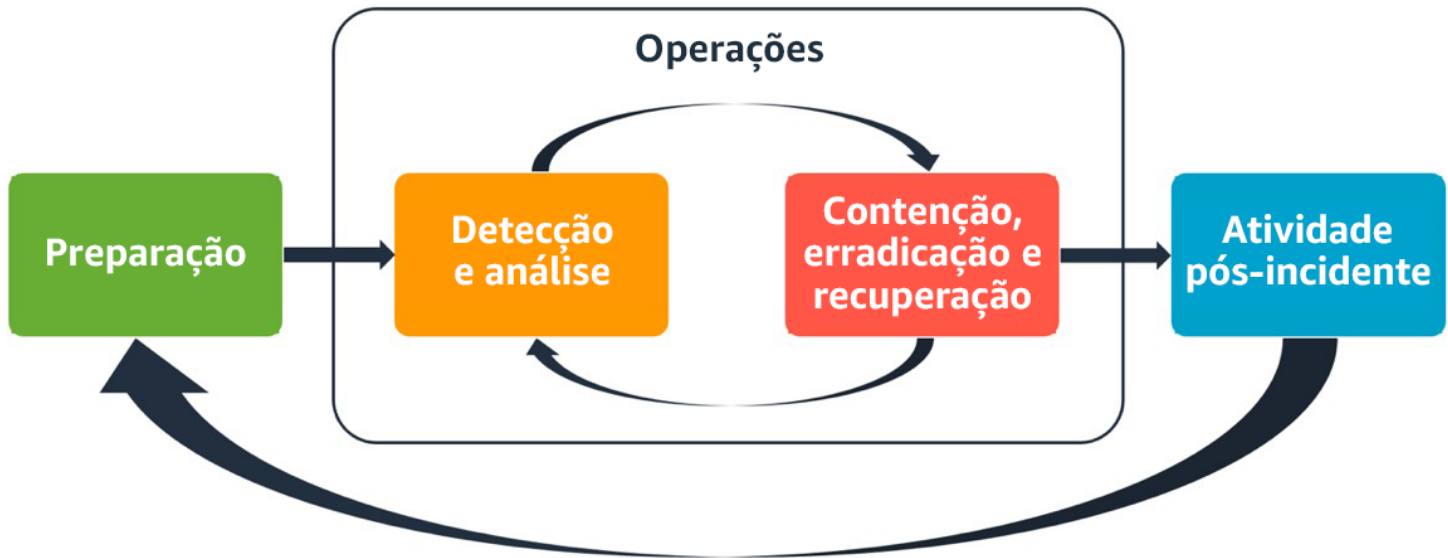
Todos os usuários da AWS de uma organização devem ter uma compreensão básica dos processos de resposta a incidentes de segurança, e a equipe de segurança deve entender como responder aos problemas de segurança. Educação, treinamento e experiência são essenciais para um programa bem-sucedido de resposta a incidentes na nuvem e são preferencialmente implementados bem antes de precisar lidar com um possível incidente de segurança. A base de um programa bem-sucedido de resposta a incidentes na nuvem é Preparação, Operações e Atividade pós-incidente.

Para entender cada um desses aspectos, considere as seguintes descrições:

- **Preparação:** prepare sua equipe de resposta a incidentes para detectar e responder aos incidentes na AWS ativando controles de detecção e verificando o acesso adequado às ferramentas e aos serviços de nuvem necessários. Além disso, prepare os manuais necessários, tanto os automatizados quanto os manuais, para garantir respostas confiáveis e consistentes.
- **Operações:** opere em eventos de segurança e possíveis incidentes seguindo as fases de resposta a incidentes do NIST: detectar, analisar, conter, erradicar e recuperar.

- **Atividade pós-incidente:** itere o resultado de seus eventos e simulações de segurança para melhorar a eficácia da resposta, aumentar o valor derivado da resposta e da investigação e reduzir ainda mais os riscos. Você precisa aprender com os incidentes e ter uma propriedade consistente das atividades de melhoria.

O diagrama a seguir mostra o fluxo desses aspectos, alinhando-se ao ciclo de vida de resposta a incidentes do NIST mencionado anteriormente, mas com operações que abrangem detecção e análise com contenção, erradicação e recuperação.



Aspectos da resposta a incidentes da AWS

Elaborar objetivos da resposta da nuvem

Embora os processos e mecanismos gerais de resposta a incidentes, como os definidos no [Guia de tratamento de incidentes de segurança de computadores NIST SP 800-61](#), continuem válidos, recomendamos que você avalie estes objetivos específicos de design que são relevantes para responder a incidentes de segurança em um ambiente de nuvem:

- **Estabelecer objetivos de resposta:** trabalhe com as partes interessadas, a assessoria jurídica e a liderança organizacional para determinar o objetivo da resposta a um incidente. Alguns objetivos comuns são: conter e atenuar o problema, recuperar os recursos afetados, preservar dados para análise forense, retornar às operações seguras conhecidas e, finalmente, aprender com os incidentes.
- **Responder usando a nuvem:** Implemente padrões de resposta na nuvem, onde o evento e os dados ocorrem.

- Saber o que você tem e do que precisa: Preserve logs, recursos, instantâneos e outras evidências copiando e armazenando-os em uma conta centralizada na nuvem dedicada à resposta. Use tags, metadados e mecanismos que impõem políticas de retenção. Você precisará entender quais serviços são usados e identificar os requisitos para investigar esses serviços. Para ajudar você a entender seu ambiente, você também pode usar a marcação.
- Usar mecanismos de reimplantação: se uma anomalia de segurança puder ser atribuída a uma configuração incorreta, a correção poderá ser tão simples quanto a remoção da variação com a reimplantação dos recursos com a configuração apropriada. Se um possível comprometimento for identificado, verifique se sua redistribuição inclui a atenuação bem-sucedida e verificada das causas principais.
- Automatizar sempre que possível: À medida que surgem problemas ou incidentes se repetem, crie mecanismos para fazer a triagem programática e responder a eventos comuns. Use respostas humanas para incidentes exclusivos, complexos ou confidenciais em que as automações são insuficientes.
- Escolher soluções escaláveis: Esforce-se para combinar a escalabilidade da abordagem de sua organização com a computação em nuvem. Implemente mecanismos de detecção e resposta que se expandam em seus ambientes para reduzir efetivamente o tempo entre a detecção e a resposta.
- Conhecer e melhorar seu processo: Seja proativo na identificação de déficits em seus processos, ferramentas ou pessoas e implemente um plano para corrigi-los. Simulações são métodos seguros para encontrar déficits e melhorar processos.

Essas metas de design são um lembrete para analisar a implementação de sua arquitetura quanto à capacidade de conduzir tanto a resposta a incidentes quanto a detecção de ameaças. Ao planejar suas implementações de nuvem, pense em responder a um incidente, de preferência, com uma metodologia de resposta sólida em termos forenses. Em alguns casos, isso significa que você pode ter várias organizações, contas e ferramentas configuradas especificamente para essas tarefas de resposta. Essas ferramentas e funções devem ser disponibilizadas para a equipe de atendimento a incidentes por meio do pipeline de implantação. Elas não devem ser estáticas, pois podem causar um risco maior.

Preparação

A preparação para um incidente é fundamental para uma resposta oportuna e eficaz a incidentes. A preparação é feita em três domínios:

- **Pessoas:** preparar seu pessoal para um incidente de segurança envolve identificar as partes interessadas relevantes para a resposta a incidentes e treiná-las em resposta a incidentes e tecnologias de nuvem.
- **Processo:** preparar seus processos para um incidente de segurança envolve documentar arquiteturas, desenvolver planos completos de resposta a incidentes e criar manuais para uma resposta consistente a eventos de segurança.
- **Tecnologia:** preparar sua tecnologia para um incidente de segurança envolve configurar o acesso, agregar e monitorar os logs necessários, implementar mecanismos de alerta eficazes e desenvolver recursos de resposta e investigação.

Cada um desses domínios é igualmente importante para uma resposta eficaz a incidentes. Nenhum programa de resposta a incidentes está completo ou é eficaz sem os três. Você precisará preparar pessoas, processos e tecnologias com uma forte integração para se preparar para um incidente.

Práticas recomendadas

- [SEC10-BP01 Identificar equipes e recursos externos fundamentais](#)
- [SEC10-BP02 Desenvolver planos de gerenciamento de incidentes](#)
- [SEC10-BP03 Prepare recursos forenses](#)
- [SEC10-BP04 Desenvolva e teste manuais de resposta a incidentes de segurança](#)
- [SEC10-BP05 Acesso pré-provisionado](#)
- [SEC10-BP06 Pré-implantação de ferramentas](#)
- [SEC10-BP07 Execute simulações](#)

SEC10-BP01 Identificar equipes e recursos externos fundamentais

Identifique as equipes, as obrigações legais e os recursos internos e externos que ajudam sua organização a responder a um incidente.

Resultado desejado: você tem uma lista das principais equipes, as respectivas informações de contato e as funções que elas desempenham ao responder a um evento de segurança. Você revisa essas informações regularmente e as atualiza para refletir mudanças de equipes do ponto de vista das ferramentas internas e externas. Você considera todos os provedores de serviços e fornecedores terceirizados ao documentar essas informações, incluindo parceiros de segurança, provedores de nuvem e aplicações de software como serviço (SaaS). Durante um evento de

segurança, as equipes estão preparadas com o nível apropriado de responsabilidade, contexto e acesso para resposta e recuperação.

Antipadrões comuns:

- Não manter uma lista atualizada das principais equipes com informações de contato, funções e responsabilidades para responder a eventos de segurança.
- Supor que todos saibam quais são as pessoas, as dependências, a infraestrutura e as soluções necessárias para resposta a um evento e recuperação.
- Não ter um documento ou repositório de conhecimentos que represente a infraestrutura principal ou o design da aplicação.
- Não ter processos de integração adequados para que novos funcionários contribuam eficazmente para uma resposta a eventos de segurança, como a realização de simulações de eventos.
- Não ter um caminho de encaminhamento estabelecido quando as equipes principais estão temporariamente indisponíveis ou não respondem durante eventos de segurança.

Benefícios do estabelecimento desta prática recomendada: esta prática reduz o tempo de triagem e resposta despendido na identificação das equipes certas e de suas funções durante um evento. Minimiza o tempo perdido durante um evento mantendo uma lista atualizada das principais equipes e de suas funções para que você possa convocar as pessoas certas para a triagem e se recuperar de um evento.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

Identifique as principais equipes da sua organização: mantenha uma lista de contatos das equipes da sua organização que você precisa envolver. Revise e atualize regularmente essas informações em caso de movimentação de equipes, como mudanças organizacionais, internas e promoções. Isso é especialmente importante para funções importantes, como gerentes de incidentes, respondentes a incidentes e líder de comunicação.

- Gerente de incidentes: a pessoa que tem autoridade geral durante a resposta a um evento.
- Respondentes a incidentes: os responsáveis pelas atividades de investigação e correção. Essas pessoas podem diferir com base no tipo de evento, mas normalmente são desenvolvedores e equipes operacionais responsáveis pela aplicação afetada.

- Líder de comunicação: a pessoa responsável pelas comunicações internas e externas, especialmente com órgãos públicos, reguladores e clientes.
- Especialistas no assunto (SMEs): no caso de equipes distribuídas e autônomas, recomendamos que você identifique um SME para workloads essenciais. Eles oferecem insights sobre a operação e a classificação de dados das workloads críticas envolvidas no evento.

Considere a possibilidade de usar o recurso [AWS Systems Manager](#) para capturar os principais contatos, definir um plano de resposta, automatizar programações de plantão e criar planos de encaminhamento. Automatize e alterne toda a equipe por meio de uma programação de plantão, para que a responsabilidade pela workload seja compartilhada entre os respectivos responsáveis. Isso favorece boas práticas, como emitir métricas e logs relevantes, bem como definir limites de alarme importantes para a workload.

Identifique parceiros externos: as empresas usam ferramentas criadas por provedores de software independentes (ISVs), parceiros e subempreiteiros para criar soluções diferenciadas para os clientes. Envolve as principais equipes dessas partes que possam ajudar na resposta e recuperação de um incidente. Recomendamos que você se inscreva no nível apropriado do AWS Support para obter acesso imediato a especialistas da AWS por meio de um caso de suporte. Considere acordos semelhantes com todos os provedores de soluções essenciais para as workloads. Alguns eventos de segurança exigem que as empresas de capital aberto notifiquem os órgãos públicos e reguladores relevantes sobre o evento e os impactos. Mantenha e atualize as informações de contato dos departamentos relevantes e das pessoas responsáveis.

Etapas da implementação

1. Configure uma solução de gerenciamento de incidentes.
 - a. Considere implantar o Incident Manager em sua conta de ferramentas de segurança.
2. Defina contatos em sua solução de gerenciamento de incidentes.
 - a. Defina pelo menos dois tipos de canal de contato para cada contato (como SMS, telefone ou e-mail) para garantir a acessibilidade durante um incidente.
3. Defina um plano de resposta.
 - a. Identifique os contatos mais adequados a serem mobilizados durante um incidente. Defina planos de encaminhamento alinhados às funções das equipes a serem mobilizadas, em vez de contatos individuais. Considere incluir contatos que possam ter a responsabilidade de informar entidades externas, mesmo que eles não sejam diretamente mobilizados para resolver o incidente.

Recursos

Práticas recomendadas relacionadas:

- [OPS02-BP03 Atividades de operações com proprietários identificados responsáveis pela performance](#)

Documentos relacionados:

- [AWS Security Incident Response Guide](#)

Exemplos relacionados:

- [AWS customer playbook framework](#)
- [Prepare for and respond to security incidents in your AWS environment](#)

Ferramentas relacionadas:

- [AWS Systems Manager Incident Manager](#)

Vídeos relacionados:

- [Amazon's approach to security during development](#)

SEC10-BP02 Desenvolver planos de gerenciamento de incidentes

O primeiro documento a ser desenvolvido para resposta a incidentes é o plano de resposta a incidentes. O plano de resposta a incidentes foi projetado para ser a base de seu programa e estratégia de resposta a incidentes.

Benefícios de estabelecer esta prática recomendada: O desenvolvimento de processos de resposta a incidentes completos e claramente definidos é fundamental para um programa de resposta a incidentes bem-sucedido e escalável. Quando ocorre um evento de segurança, etapas e fluxos de trabalho claros poderão ajudar você a responder em tempo hábil. Talvez você já tenha processos de resposta a incidentes existentes. Independentemente do seu estado atual, é importante atualizar, repetir e testar seus processos de resposta a incidentes regularmente.

Nível de risco exposto se esta prática recomendada não for estabelecida: alto

Orientação para implementação

Um plano de gerenciamento de incidentes é fundamental para responder, mitigar e se recuperar de possíveis impactos de incidentes de segurança. Um plano de gerenciamento de incidentes é um processo estruturado de identificação, correção e resposta em tempo hábil a incidentes de segurança.

A nuvem tem muitos dos mesmos requisitos e perfis operacionais encontrados em um ambiente on-premises. Ao criar um plano de gerenciamento de incidentes, é importante definir estratégias de resposta e recuperação que se alinhem melhor aos seus resultados empresariais e requisitos de conformidade. Por exemplo, se você opera workloads na AWS em conformidade com o FedRAMP nos Estados Unidos, é útil aderir ao [Guia de tratamento de segurança de computadores NIST SP 800-61](#). Da mesma forma, ao operar workloads com informações de identificação pessoal (PII) da Europa, considere cenários como a maneira como você deve se proteger e responder a incidentes relacionados à residência de dados, conforme exigido pela [Regulamentação Geral de Proteção de Dados \(GDPR\) da UE](#).

Ao criar um plano de gerenciamento de incidentes para suas workloads na AWS, comece com o [Modelo de responsabilidade compartilhada da AWS](#), para elaborar uma abordagem de defesa profunda em relação à resposta a incidentes. Nesse modelo, a AWS gerencia a segurança da nuvem, e você é responsável pela segurança na nuvem. Isso significa que você mantém o controle e é responsável pelos controles de segurança que escolhe implementar. O [AWS Security Incident Response Guide \(Guia de resposta a incidentes de segurança da AWS\)](#) detalha os conceitos e as orientações básicas para criar um plano de gerenciamento de incidentes centrado na nuvem.

Um plano de gerenciamento de incidentes eficaz deve ser continuamente iterado e permanecer atualizado com relação às suas metas de operações de nuvem. Considere o uso dos planos de implementação detalhados abaixo, à medida que cria e evolui seu plano de gerenciamento de incidentes.

Etapas da implementação

Defina funções e responsabilidades

Lidar com eventos de segurança exige disciplina interorganizacional e uma inclinação para a ação. Em sua estrutura organizacional, deve haver muitas pessoas responsáveis, atribuídas, consultadas ou mantidas informadas durante um incidente, como representantes de recursos humanos (RH), da equipe executiva e do setor jurídico. Considere essas funções e responsabilidades e se algum terceiro deve estar envolvido. Observe que muitas regiões têm leis locais que regem o que deve e o

que não deve ser feito. Embora possa parecer burocrático criar um grafo de pessoas responsáveis, atribuídas, consultadas e informadas (RACI) para seus planos de resposta de segurança, isso facilita a comunicação rápida e direta e descreve claramente a liderança em diferentes estágios do evento.

Durante um incidente, incluir os proprietários e os desenvolvedores de aplicações e recursos afetados é fundamental porque eles são especialistas no assunto (PMEs) que podem fornecer informações e contexto para ajudar a medir o impacto. Pratique e construa relacionamentos com os desenvolvedores e os proprietários de aplicações antes de confiar na experiência deles para responder a incidentes. Proprietários de aplicações ou PMEs, como administradores ou engenheiros de nuvem, podem precisar agir em situações em que o ambiente não seja familiar ou tenha complexidade, ou em que os respondentes não tenham acesso.

Por fim, parceiros confiáveis podem estar envolvidos na investigação ou na resposta, pois podem oferecer experiência adicional e um controle valioso. Se você não tiver essas habilidades em sua própria equipe, contrate uma parte externa para obter assistência.

Entender as equipes de resposta e o suporte da AWS

- AWS Support
 - [O AWS Support](#) oferece uma variedade de planos que concedem acesso a ferramentas e conhecimentos que apoiam o êxito e a saúde operacional de suas soluções da AWS. Se precisar de suporte técnico e mais recursos para ajudar a planejar, implantar e otimizar seu ambiente da AWS, selecione um plano de suporte mais adequado ao seu caso de uso da AWS.
 - Considere o [Support Center](#) entre AWS Management Console (é necessário fazer login) como ponto central de contato para obter suporte para problemas que afetam seus recursos da AWS. O acesso ao AWS Support é controlado pelo AWS Identity and Access Management. Para ter mais informações sobre como obter acesso aos recursos da AWS Support, consulte [Conceitos básicos do AWS Support](#).
- Equipe de Resposta a Incidentes de Clientes (CIRT) da AWS
 - A Equipe de Resposta a Incidentes de Clientes (CIRT) da AWS é uma equipe global da AWS especializada 24 horas por dia, 7 dias por semana, que presta assistência aos clientes durante eventos de segurança ativos do cliente do [Modelo de responsabilidade compartilhada da AWS](#).
 - Ao apoiar você, a AWS CIRT presta assistência na triagem e na recuperação de um evento de segurança ativo na AWS. Eles podem ajudar na análise da causa raiz por meio do uso de logs de serviço da AWS e fornecer recomendações para recuperação. Eles também podem fornecer recomendações de segurança e práticas recomendadas para ajudar você a evitar eventos de segurança no futuro.

- Os clientes da AWS podem contratar a AWS CIRT por meio de um [caso do AWS Support](#).
- Suporte de resposta a DDoS
 - A AWS oferece o [AWS Shield](#), que fornece um serviço gerenciado de proteção distribuída de negação de serviço (DDoS) que protege as aplicações web em execução na AWS. O Shield oferece detecção contínua e mitigações automáticas em linha que podem minimizar o tempo de inatividade e a latência da aplicação, portanto, não há necessidade de contratar o AWS Support para se beneficiar da proteção contra DDoS. Existem dois níveis de Shield: AWS Shield Standard e AWS Shield Advanced. Para saber mais sobre as diferenças entre esses dois níveis, consulte a [documentação de recursos do Shield](#).
- AWS Managed Services (AMS)
 - [O AWS Managed Services \(AMS\)](#) oferece gerenciamento contínuo de sua infraestrutura da AWS para que você possa se concentrar em suas aplicações. Ao implementar as práticas recomendadas para manter sua infraestrutura, o AMS ajuda a reduzir sua sobrecarga operacional e os riscos. O AMS automatiza atividades comuns, como solicitações de mudança, monitoramento, gerenciamento de patches, serviços de segurança e backup, e fornece serviços de ciclo de vida completo para provisionar, executar e oferecer compatibilidade com sua infraestrutura.
 - O AMS assume a responsabilidade de implantar um pacote de controles de detecção de segurança e fornece uma primeira linha de resposta aos alertas 24 horas por dia, 7 dias por semana. Quando um alerta é iniciado, o AMS segue um conjunto padrão de guias e manuais automatizados para verificar uma resposta consistente. Esses guias são compartilhados com os clientes do AMS durante a integração para que eles possam desenvolver e coordenar uma resposta com o AMS.

Desenvolva o plano de resposta a incidentes

O plano de resposta a incidentes foi projetado para ser a base de seu programa e estratégia de resposta a incidentes. O plano de resposta a incidentes deve estar em um documento formal. Um plano de resposta a incidentes geralmente inclui as seguintes seções:

- Uma visão geral da equipe de resposta a incidentes: Descreve as metas e as funções da equipe de resposta a incidentes.
- Funções e responsabilidades: Lista as partes interessadas na resposta a incidentes e detalha suas funções quando ocorre um incidente.

- Um plano de comunicação: Detalha as informações de contato e como você se comunica durante um incidente.
- Métodos de comunicação de backup: É prática recomendada ter a comunicação fora de banda como backup para a comunicação de incidentes. Um exemplo de aplicação que fornece um canal seguro de comunicação fora de banda é AWS Wickr.
- Fases da resposta a incidentes e ações a serem realizadas: Enumera as fases da resposta a incidentes (por exemplo, detectar, analisar, erradicar, conter e recuperar), incluindo ações de alto nível a serem realizadas nessas fases.
- Definições de severidade e priorização do incidente: Detalha como classificar a severidade de um incidente, como priorizar o incidente e, depois, como as definições de severidade afetam os procedimentos de escalonamento.

Embora essas seções sejam comuns em empresas de diferentes tamanhos e setores, o plano de resposta a incidentes de cada organização é único. Você precisa criar um plano de resposta a incidentes que funcione melhor para a organização.

Recursos

Práticas recomendadas relacionadas:

- [SEC04 \(Como você detecta e investiga eventos de segurança?\)](#)

Documentos relacionados:

- [AWS Security Incident Response Guide \(Guia de resposta a incidentes de segurança da AWS\)](#)
- [NIST: Guia de tratamento de incidentes de segurança de computadores](#)

SEC10-BP03 Prepare recursos forenses

Antes de um incidente de segurança, considere o desenvolvimento de recursos forenses para apoiar as investigações de eventos de segurança.

Nível de risco exposto se esta prática recomendada não for estabelecida: Médio

Os conceitos da análise forense on-premises tradicional se aplicam à AWS. Para obter informações importantes para começar a desenvolver recursos forenses na Nuvem AWS, consulte [Forensic investigation environment strategies in the Nuvem AWS](#).

Depois de configurar o ambiente e a estrutura da Conta da AWS para análise forense, defina as tecnologias necessárias para executar com eficácia metodologias forenses sólidas nas quatro fases:

- **Coleta:** Colete logs relevantes da AWS, como logs do AWS CloudTrail, do AWS Config, logs de fluxo da VPC e log em nível de host. Colete snapshots, backups e despejos de memória dos recursos afetados da AWS, quando disponíveis.
- **Exame:** Examine os dados coletados extraíndo e avaliando as informações relevantes.
- **Análises:** Analise os dados coletados para entender o incidente e tirar conclusões dele.
- **Relatórios:** Apresente as informações resultantes da fase de análise.

Etapas da implementação

Prepare o ambiente forense

[AWS Organizations](#) ajuda a gerenciar e reger centralmente um ambiente da AWS à medida que você expande e escala os recursos da AWS. Uma organização da AWS consolida suas Contas da AWS para que você possa administrá-las como uma única unidade. Você pode usar unidades organizacionais (UOs) para agrupar contas e administrá-las como uma única unidade.

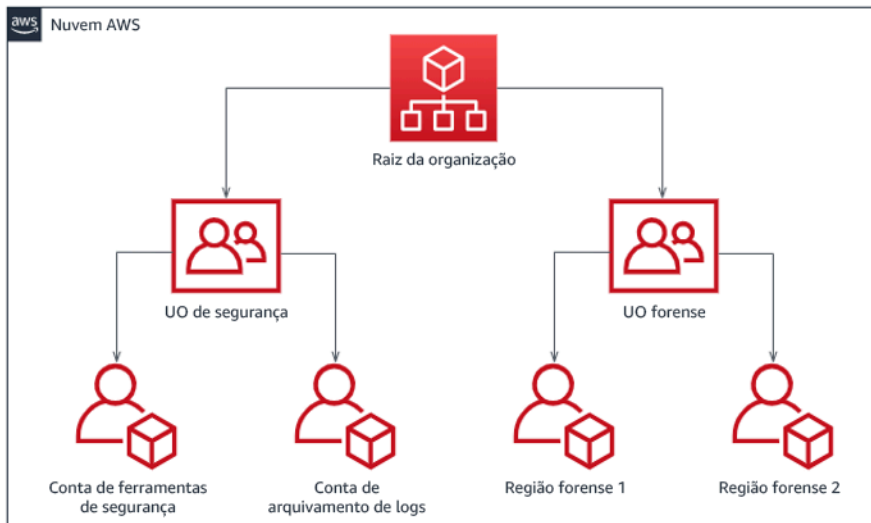
Para resposta a incidentes, é útil ter uma estrutura da Conta da AWS compatível com as funções de resposta a incidentes, que inclui uma UO de segurança e uma UO forense. Dentro da OU de segurança, você deve ter contas para:

- **Arquivamento de logs:** Agregue logs em uma Conta da AWS de arquivamento de logs com permissões limitadas.
- **Ferramentas de segurança:** Centralize os serviços de segurança em uma Conta da AWS de ferramenta de segurança. Essa conta opera como administrador delegado dos serviços de segurança.

Dentro da UO forense, você tem a opção de implementar uma única conta ou contas forenses para cada região em que opera, dependendo da que funciona melhor para sua empresa e modelo operacional. Se você criar uma conta forense por região, poderá bloquear a criação de recursos da AWS fora dessa região e reduzir o risco de os recursos serem copiados para uma região não pretendida. Por exemplo, se você opera apenas em US East (N. Virginia) Region (us-east-1) e US West (Oregon) (us-west-2), então você teria duas contas na UO forense: uma para us-east-1 e uma para us-west-2.

Você pode criar uma Conta da AWS de análise forense para várias regiões. Você deve ter cuidado ao copiar recursos da AWS para essa conta para verificar se está de acordo com seus requisitos de soberania de dados. Como é preciso tempo para provisionar novas contas, é imperativo criar e instrumentar as contas forenses bem antes de um incidente, para que os respondentes possam estar preparados para usá-las de forma eficaz em suas respostas.

O diagrama a seguir exibe um exemplo de estrutura de contas, incluindo uma UO forense com contas forenses por região:



Estrutura de contas por região para resposta a incidentes

Capture backups e snapshots

Configurar backups dos principais sistemas e bancos de dados é essencial para a recuperação de um incidente de segurança e para fins forenses. Com os backups em vigor, você pode restaurar seus sistemas ao estado seguro anterior. Na AWS, você pode criar snapshots de vários recursos. Os snapshots fornecem backups pontuais desses recursos. Há muitos serviços da AWS que podem ajudar em backup e recuperação. Para obter detalhes sobre esses serviços e abordagens para backup e recuperação, consulte [Backup and Recovery Prescriptive Guidance](#) e [Use backups to recover from security incidents](#).

Especialmente quando se trata de situações como ransomware, é fundamental que os backups estejam bem protegidos. Para obter orientações sobre como proteger os backups, consulte [Top 10 security best practices for securing backups in AWS](#). Além de proteger os backups, você deve testar regularmente seus processos de backup e restauração para verificar se a tecnologia e os processos implementados funcionam conforme o esperado.

Automatize a análise forense

Durante um evento de segurança, sua equipe de resposta a incidentes deve ser capaz de coletar e analisar evidências rapidamente, mantendo a precisão durante o período em torno do evento (como capturar registros relacionados a um evento ou recurso específico ou coletar o despejo de memória de uma instância do Amazon EC2). É desafiador e demorado para a equipe de resposta a incidentes coletar manualmente as evidências relevantes, especialmente em um grande número de instâncias e contas. Além disso, a coleta manual pode estar sujeita a erros humanos. Por esses motivos, você deve desenvolver e implementar a automação para perícia o máximo possível.

A AWS oferece vários recursos de automação para análise forense, que estão listados na seção de recursos a seguir. Esses recursos são exemplos de padrões forenses que desenvolvemos e que os clientes implementaram. Embora possam ser uma arquitetura de referência útil para começar, considere modificá-las ou criar padrões de automação forense com base em seu ambiente, requisitos, ferramentas e processos forenses.

Recursos

Documentos relacionados:

- [AWS Security Incident Response Guide - Develop Forensics Capabilities](#)
- [AWS Security Incident Response Guide - Forensics Resources](#)
- [Forensic investigation environment strategies in the Nuvem AWS](#)
- [How to automate forensic disk collection in AWS](#)
- [AWS Prescriptive Guidance - Automate incident response and forensics](#)

Vídeos relacionados:

- [Automatização de resposta a incidentes e forense](#)

Exemplos relacionados:

- [Automated Incident Response and Forensics Framework \(Estrutura forense e de resposta automatizada a incidentes\)](#)
- [Automated Forensics Orchestrator for Amazon EC2](#)

SEC10-BP04 Desenvolva e teste manuais de resposta a incidentes de segurança

Uma parte fundamental da preparação de seus processos de resposta a incidentes é desenvolver manuais. Os manuais de resposta a incidentes fornecem uma série de orientações prescritivas e etapas a serem seguidas quando ocorre um evento de segurança. Ter uma estrutura e etapas claras simplifica a resposta e reduz a probabilidade de erro humano.

Nível de risco exposto se esta prática recomendada não for estabelecida: Médio

Orientação para implementação

Os manuais devem ser criados para cenários de incidentes, como:

- Incidentes esperados: os manuais devem ser criados para os incidentes previstos. Isso inclui ameaças como negação de serviço (DoS), ransomware e comprometimento de credenciais.
- Descobertas ou alertas de segurança conhecidos: os manuais devem ser criados para descobertas e alertas de segurança conhecidos, como descobertas do GuardDuty. Você pode receber uma descoberta do GuardDuty e pensar: “E agora?” Para evitar que você trate incorretamente ou ignore uma descoberta do GuardDuty, crie um manual para cada possível descoberta do GuardDuty. Alguns detalhes e orientações sobre a correção podem ser encontrados na [documentação do GuardDuty](#). É importante notar que o GuardDuty não está habilitado por padrão e tem um custo. Para obter mais detalhes sobre o GuardDuty, consulte [Appendix A: Cloud capability definitions - Visibility and alerting](#).

Os manuais devem conter etapas técnicas a serem concluídas por um analista de segurança para investigar e responder adequadamente a um possível incidente de segurança.

Etapas da implementação

Os itens a serem incluídos em um manual incluem:

- Visão geral do manual: qual cenário de risco ou incidente esse manual aborda? Qual é o objetivo do manual?
- Pré-requisitos: quais logs, mecanismos de detecção e ferramentas automatizadas são necessários para esse cenário de incidente? Qual é a notificação esperada?
- Informações de comunicação e escalonamento: quem está envolvido e quais são suas informações de contato? Quais são as responsabilidades de cada parte interessada?

- Etapas de resposta: em todas as fases da resposta a incidentes, quais etapas táticas devem ser seguidas? Quais consultas um analista deve executar? Qual código deve ser executado para alcançar o resultado desejado?
 - Detectar: como o incidente será detectado?
 - Análise: como o escopo do impacto será determinado?
 - Contêm: como o incidente será isolado para limitar o escopo?
 - Erradicar: como a ameaça será removida do ambiente?
 - Recuperar: como o sistema ou o recurso afetado voltará à produção?
- Resultados esperados: depois que as consultas e o código forem executados, qual é o resultado esperado do manual?

Recursos

Práticas recomendadas relacionadas ao Well-Architected:

- [SEC10-BP02 - Develop incident management plans \(SEC10-BP02 – Desenvolver planos de gerenciamento de incidentes\)](#)

Documentos relacionados:

- [Framework for Incident Response Playbooks \(Estrutura para manuais de resposta a incidentes\)](#)
- [Develop your own Incident Response Playbooks \(Desenvolva seus próprios manuais de resposta a incidentes\)](#)
- [Incident Response Playbook Samples \(Amostras do manual de resposta a incidentes\)](#)
- [Building an AWS incident response runbook using Jupyter playbooks and CloudTrail Lake](#)

SEC10-BP05 Acesso pré-provisionado

Verifique se os respondentes a incidentes têm o acesso correto pré-provisionado na AWS para reduzir o tempo de investigação necessário até a recuperação.

Antipadrões comuns:

- Uso da conta raiz para a resposta a incidentes.
- Alteração de contas de usuário existentes.

- Manipulação de permissões do IAM diretamente ao fornecer elevação de privilégios just-in-time.

Nível de risco exposto se essa prática recomendada não for estabelecida: Médio

Orientação para implementação

A AWS recomenda reduzir ou eliminar a dependência de credenciais de longa duração sempre que possível, dando preferência a credenciais temporárias e a mecanismos de escalação de privilégios just-in-time. As credenciais de longa duração são propensas a riscos de segurança e aumentam a sobrecarga operacional. Para a maioria das tarefas de gerenciamento, bem como tarefas de resposta a incidentes, recomendamos a implementação da [federação de identidades](#) junto com a [escalação temporária para acesso administrativo](#). Nesse modelo, um usuário solicita elevação a um nível superior de privilégio (como um perfil de resposta a incidentes) e, considerando que ele seja elegível para a elevação, a solicitação é enviada a um aprovador. Se a solicitação for aprovada, o usuário receberá um conjunto de credenciais [temporárias da AWS](#), que podem ser usadas para concluir suas tarefas. Depois que essas credenciais expirarem, o usuário deve enviar uma nova solicitação de elevação.

Recomendamos o uso da escalação de privilégio temporária para a maioria dos cenários de resposta a incidentes. A maneira correta de fazer isso é com o uso do [AWS Security Token Service](#) e [de políticas de sessão](#) para definir o escopo de acesso.

Há cenários em que as identidades federadas não estão disponíveis, como:

- Interrupção relacionada a um provedor de identidades (IdP) comprometido.
- Erro de configuração ou erro humano causando uma falha no sistema de gerenciamento de acesso federado.
- Atividade mal-intencionada, como um evento de negação de serviço distribuído (DDoS) ou indisponibilidade de renderização do sistema.

Nos casos anteriores, deverá haver um acesso de emergência de breaking-glass configurado para permitir a investigação e a correção em tempo hábil dos incidentes. Recomendamos a utilização de um [usuário do IAM com as permissões apropriadas](#) para realizar tarefas e acessar os recursos da AWS. Use as credenciais raiz somente para [tarefas que exijam o acesso do usuário raiz](#). Para verificar se os respondentes de um incidente têm o nível de acesso correto à AWS e a outros sistemas relevantes, recomendamos o pré-provisionamento de contas de usuário dedicadas. As contas de usuário exigem acesso privilegiado e devem ser estritamente controladas

e monitoradas. As contas devem ser criadas com os menores privilégios exigidos para realizar as tarefas necessárias, e o nível de acesso deve ser baseado nos manuais criados como parte do plano de gerenciamento de incidentes.

Utilize perfis e usuários dedicados e com propósito específico como uma prática recomendada. Escalar temporariamente o acesso de usuários ou perfis por meio da adição de políticas do IAM não deixa claro qual é o acesso que os usuários tinham durante o incidente, e há um risco de que os privilégios escalados não sejam revogados.

É importante remover o máximo de dependências possível para verificar se o acesso pode ser obtido com o maior número possível de cenários de falha. Para apoiar isso, crie um manual para verificar se os usuários de resposta a incidentes são criados como usuários do AWS Identity and Access Management em uma conta de segurança dedicada, e não são gerenciados por nenhuma solução de autenticação única (SSO) ou federação. Cada respondente individual deve ter sua própria conta nomeada. A configuração da conta deve aplicar uma [política de senha forte](#) e a autenticação multifator (MFA). Se os manuais de resposta a incidentes só exigem acesso ao AWS Management Console, o usuário não deve ter chaves de acesso configuradas e deve ser proibido explicitamente de criar chaves de acesso. Isso pode ser configurado com políticas do IAM ou políticas de controle de serviços (SCPs), conforme mencionado nas Práticas recomendadas de segurança da AWS para [SCPs do AWS Organizations](#). Os usuários não devem ter privilégios além da capacidade de assumir perfis de resposta a incidentes em outras contas.

Durante um incidente, pode ser necessário conceder acesso a outros indivíduos internos ou externos para apoiar a investigação, a correção ou as atividades de recuperação. Nesse caso, use o mecanismo do manual mencionado anteriormente, e deve haver um processo para verificar se qualquer acesso adicional foi revogado imediatamente após a conclusão do incidente.

Para verificar se o uso de perfis de resposta a incidentes pode ser monitorado e auditado corretamente, é essencial que as contas de usuário do IAM criadas para esse fim não sejam compartilhadas entre indivíduos e que o usuário raiz da Conta da AWS não seja utilizado, a menos que isso seja [exigido para uma tarefa específica](#). Se o usuário raiz for exigido (por exemplo, quando o acesso do IAM a uma conta específica estiver indisponível), use um processo distinto com um manual disponível para verificar a disponibilidade da senha e do token de MFA do usuário raiz.

Para configurar as políticas do IAM para os perfis de resposta a incidentes, considere o uso do [IAM Access Analyzer](#) para gerar políticas baseadas em logs do AWS CloudTrail. Para fazer isso, conceda acesso de administrador ao perfil de resposta a incidentes em uma conta de não produção e execute de acordo com os manuais. Depois da conclusão, pode ser criada uma política que permita somente as ações realizadas. Essa política pode ser então aplicada a todos os perfis de resposta a incidentes

em todas as contas. Você pode criar uma política do IAM separada para cada manual a fim de facilitar o gerenciamento e a auditoria. Exemplos de manuais podem incluir planos de resposta para ransomware, violações de dados, perda de acesso da produção, dentre outros cenários.

Use as contas de usuário de resposta a incidentes para assumir funções do [IAM de resposta a incidentes em outras Contas da AWS](#). Esses perfis também devem ser configurados para só poderem ser assumidos por usuários na conta de segurança, e o relacionamento de confiança deve exigir que a entidade principal que está fazendo a chamada seja autenticada com MFA. Os perfis devem usar políticas do IAM com escopo estritamente definido para controlar o acesso. Garanta que todas as solicitações `AssumeRole` para esses perfis estejam conectadas no CloudTrail e sejam alertadas, e que as ações realizadas usando esses perfis sejam registradas.

É altamente recomendável que as contas de usuário do IAM e os perfis do IAM sejam claramente nomeados para permitir que sejam encontrados com facilidade nos logs do CloudTrail. Um exemplo disso seria nomear as contas do IAM como `<USER_ID>-BREAK-GLASS` e os perfis do IAM como `BREAK-GLASS-ROLE`.

[O CloudTrail](#) é usado para registrar as atividades da API em suas contas da AWS e deve ser usado para [configurar alertas sobre o uso dos perfis de resposta a incidentes](#). Consulte a publicação do blog sobre como configurar alertas quando as chaves raiz são usadas. As instruções podem ser modificadas para configurar a métrica do [Amazon CloudWatch](#) filtro a filtro em eventos `AssumeRole` relacionados ao perfil do IAM de resposta a incidentes:

```
{ $.eventName = "AssumeRole" && $.requestParameters.roleArn =  
  "<INCIDENT_RESPONSE_ROLE_ARN>" && $.userIdentity.invokedBy NOT EXISTS && $.eventType !=  
  "AwsServiceEvent" }
```

Como é provável que os perfis de resposta a incidentes tenham um alto nível de acesso, é importante que esses alertas sejam transmitidos a um grande grupo e que sejam tomadas atitudes rapidamente.

Durante um incidente, é possível que um respondente possa exigir acesso a sistemas que não são protegidos diretamente pelo IAM. Isso pode incluir instâncias do Amazon Elastic Compute Cloud, bancos de dados do Amazon Relational Database Service ou plataformas de software como serviço (SaaS). É altamente recomendável que, em vez de usar protocolos nativos, como SSH ou RDP, o [AWS Systems Manager Session Manager](#) seja usado para todo acesso administrativo a instâncias do Amazon EC2. Esse acesso pode ser controlado usando o IAM, que é protegido e auditado. Também pode ser possível automatizar partes de seus manuais usando os documentos do [AWS Systems Manager Run Command](#), o que pode reduzir os erros do usuário e melhorar o tempo

de recuperação. Para acesso aos bancos de dados e a ferramentas de terceiros, recomendamos armazenar as credenciais de acesso no AWS Secrets Manager e conceder acesso aos perfis de respondente a incidentes.

Por fim, o gerenciamento das contas de usuário do IAM de resposta a incidentes deve ser adicionado aos seus processos de [junção, migração e saída](#), além de ser revisado e testado periodicamente visando confirmar se somente o acesso pretendido é permitido.

Recursos

Documentos relacionados:

- [Managing temporary elevated access to your AWS environment \(Gerenciamento de acesso elevado temporário ao seu ambiente da AWS\)](#)
- [AWS Security Incident Response Guide \(Guia de resposta a incidentes de segurança da AWS\)](#)
- [AWS Elastic Disaster Recovery](#)
- [AWS Systems Manager Incident Manager](#)
- [Setting an account password policy for IAM users \(Definição de uma política de senhas de contas para usuários do IAM\)](#)
- [Using multi-factor authentication \(MFA\) in AWS \(Uso da autenticação multifator \(MFA\) na AWS\)](#)
- [Configuring Cross-Account Access with MFA \(Configuração do acesso entre contas com MFA\)](#)
- [Using IAM Access Analyzer to generate IAM policies \(Uso do IAM Access Analyzer para gerar políticas do IAM\)](#)
- [Best Practices for AWS Organizations Service Control Policies in a Multi-Account Environment \(Práticas recomendadas para políticas de controle de serviço do AWS Organizations em um ambiente de várias contas\)](#)
- [How to Receive Notifications When Your AWS Account's Root Access Keys Are Used \(Como receber notificações quando as chaves de acesso raiz da sua conta da AWS são usadas\)](#)
- [Create fine-grained session permissions using IAM managed policies \(Criar permissões de sessão refinadas usando políticas gerenciadas pelo IAM\)](#)

Vídeos relacionados:

- [Automating Incident Response and Forensics in AWS \(Automação de resposta a incidentes e investigações forenses na AWS\)](#)
- [Guia DIY \(faça você mesmo\) para runbooks, relatórios de incidentes e resposta a incidentes](#)

- [Prepare for and respond to security incidents in your AWS environment \(Prepare-se e responda a incidentes de segurança no ambiente da AWS\)](#)

Exemplos relacionados:

- [Lab: AWS Account Setup and Root User \(Laboratório: usuário raiz e configuração de conta da AWS\)](#)
- [Lab: Incident Response with AWS Console and CLI \(Laboratório: resposta a incidentes com o console e a CLI da AWS\)](#)

SEC10-BP06 Pré-implantação de ferramentas

Verifique se o pessoal de segurança tem as ferramentas certas pré-implantadas para reduzir o tempo de investigação até a recuperação.

Nível de risco exposto se esta prática recomendada não for estabelecida: Médio

Orientação para implementação

Para automatizar as funções de resposta e operações de segurança, você pode usar um conjunto abrangente de APIs e ferramentas da AWS. Você pode automatizar totalmente os recursos de gerenciamento de identidade, segurança de rede, proteção de dados e monitoramento e disponibilizá-los com métodos populares de desenvolvimento de software já em vigor. Quando você cria a automação da segurança, seu sistema pode monitorar, analisar e iniciar uma resposta, em vez de fazer com que as pessoas monitorem a sua posição de segurança e reajam manualmente a eventos.

Se as equipes de resposta a incidentes continuarem a responder aos alertas da mesma forma, há o risco de se acostumarem aos alertas. Com o passar do tempo, a equipe pode se tornar dessensibilizada para alertas e cometer erros ao lidar com situações comuns ou perder alertas incomuns. A automação ajuda a evitar a exaustão de alertas usando funções que processam alertas repetitivos e comuns, permitindo que as pessoas lidem com incidentes confidenciais e exclusivos. A integração de sistemas de detecção de anomalias, como Amazon GuardDuty, AWS CloudTrail Insights e Amazon CloudWatch Anomaly Detection, pode reduzir a carga de alertas baseados em limites comuns.

Você pode melhorar os processos manuais com a automatização programática das etapas do processo. Depois de definir o padrão de correção para um evento, você pode decompor esse padrão

em lógica acionável e desenvolver o código para executar essa lógica. Os respondentes podem executar esse código para corrigir o problema. Com o passar do tempo, você pode automatizar mais e mais etapas e, por fim, lidar automaticamente com classes inteiras de incidentes comuns.

Durante uma investigação de segurança, você precisa ser capaz de analisar os logs relevantes para registrar e compreender o escopo completo e o cronograma do incidente. Os logs também são necessários para geração de alertas indicando que ocorreram determinadas ações de interesse. É essencial selecionar, ativar, armazenar e configurar mecanismos de consulta, recuperação e definir alertas. Além disso, uma forma eficaz de fornecer ferramentas para pesquisar dados de log é o [Amazon Detective](#).

A AWS oferece mais de 200 serviços em nuvem e milhares de recursos. Recomendamos que você analise os serviços que podem apoiar e simplificar sua estratégia de resposta a incidentes.

Além do registro em log, você deve desenvolver e implementar uma estratégia [consistente de marcação](#). A marcação pode ajudar a fornecer contexto sobre a finalidade de um recurso da AWS. A marcação também pode ser usada para automação.

Etapas da implementação

Selecione e configure logs para análise e alertas

Consulte a documentação a seguir sobre como configurar logs para resposta a incidentes:

- [Logging strategies for security incident response \(Estratégias de registro para resposta a incidentes de segurança\)](#)
- [SEC04-BP01 Configurar registro em log de serviço e aplicação](#)

Habilite serviços de segurança para oferecer suporte à detecção e resposta

A AWS fornece recursos nativos de detecção, prevenção e resposta, e outros serviços podem ser usados para arquitetar soluções de segurança personalizadas. Para obter uma lista dos serviços mais relevantes para resposta a incidentes de segurança, consulte [Definições de capacidade de nuvem](#).

Desenvolva e implemente uma estratégia de marcação

Obter informações contextuais sobre o caso de uso empresarial e as partes interessadas internas relevantes em torno de um recurso da AWS pode ser difícil. Uma forma de fazer isso é na forma

de tags, que atribuem metadados aos recursos da AWS e consistem em uma chave e um valor definidos pelo usuário. Você pode criar tags para categorizar os recursos por finalidade, proprietário, ambiente, tipo de dados processados e outros critérios de sua escolha.

Ter uma estratégia de marcação consistente pode acelerar os tempos de resposta e minimizar o tempo gasto no contexto organizacional, permitindo identificar e discernir rapidamente as informações contextuais sobre um recurso da AWS. As tags também podem servir como um mecanismo para iniciar automações de resposta. Para obter mais detalhes sobre o que marcar, consulte [Tagging your AWS resources](#). Primeiro, você deve definir as tags que deseja implementar em toda a sua organização. Depois disso, você implementará e aplicará essa estratégia de marcação. Para obter mais detalhes sobre implementação e aplicação, consulte [Implement AWS resource tagging strategy using AWS Tag Policies and Service Control Policies \(SCPs\)](#).

Recursos

Práticas recomendadas relacionadas ao Well-Architected:

- [SEC04-BP01 Configurar registro em log de serviço e aplicação](#)
- [SEC04-BP02 Capturar logs, descobertas e métricas em locais padronizados](#)

Documentos relacionados:

- [Logging strategies for security incident response \(Estratégias de registro para resposta a incidentes de segurança\)](#)
- [Incident response cloud capability definitions \(Definições de recursos de nuvem de resposta a incidentes\)](#)

Exemplos relacionados:

- [Threat Detection and Response with Amazon GuardDuty and Amazon Detective](#)
- [Security Hub Workshop \(Workshop do Security Hub\)](#)
- [Vulnerability Management with Amazon Inspector](#)

SEC10-BP07 Execute simulações

À medida que as organizações crescem e evoluem com o tempo, o mesmo acontece com o cenário de ameaças, o que torna importante analisar continuamente seus recursos de resposta a incidentes.

Executar simulações (também conhecidas como dias de teste) é um método que pode ser usado para realizar essa avaliação. As simulações usam cenários de eventos de segurança do mundo real projetados para imitar as táticas, as técnicas e os procedimentos (TTPs) de um agente de ameaças e permitir que uma organização exercite e avalie seus recursos de resposta a incidentes respondendo a esses eventos cibernéticos simulados da mesma forma que em uma situação real.

Benefícios do estabelecimento dessa prática recomendada: as simulações têm vários benefícios:

- Validar a prontidão cibernética e desenvolver a confiança de seus socorristas.
- Testar a precisão e a eficiência de ferramentas e fluxos de trabalho.
- Refinar os métodos de comunicação e escalonamento alinhados com seu plano de resposta a incidentes.
- Proporcionar uma oportunidade de responder a vetores menos comuns.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Existem três tipos principais de simulações:

- **Simulações teóricas:** a abordagem de simulações teóricas é uma sessão baseada em discussões que envolvem as várias partes interessadas na resposta a incidentes para exercer funções e responsabilidades e usar ferramentas de comunicação e manuais estabelecidos. A facilitação das simulações normalmente pode ser realizada em um dia inteiro em um local virtual, local físico ou uma combinação de ambos. Por ser baseada em discussões, a simulação teórica se concentra em processos, pessoas e colaboração. A tecnologia é parte integrante da discussão, mas o uso real de ferramentas ou scripts de resposta a incidentes geralmente não faz parte da simulação teórica.
- **Simulações da equipe roxa:** as simulações da equipe roxa aumentam o nível de colaboração entre os respondentes ao incidente (equipe azul) e os agentes de ameaças simuladas (equipe vermelha). A equipe azul é composta por membros do centro de operações de segurança (SOC), mas também pode incluir outras partes interessadas que estariam envolvidas durante um evento cibernético real. A equipe vermelha é composta por uma equipe de testes de penetração ou pelas principais partes interessadas treinadas em segurança ofensiva. A equipe vermelha trabalha em colaboração com os facilitadores da simulação ao projetar um cenário para que este seja preciso e viável. Durante as simulações da equipe roxa, o foco principal está nos mecanismos de detecção, nas ferramentas e nos procedimentos operacionais padrão (SOPs) que apoiam os esforços de resposta a incidentes.

- Simulações da equipe vermelha: durante uma simulação da equipe vermelha, o infrator (equipe vermelha) realiza uma simulação para atingir um determinado objetivo ou conjunto de objetivos a partir de um escopo predeterminado. Os defensores (equipe azul) não necessariamente terão conhecimento do escopo e da duração da simulação, o que oferece uma avaliação mais realista de como eles responderiam a um incidente real. Como as simulações da equipe vermelha podem ser testes invasivos, tenha cuidado e implemente controles para verificar se a simulação não causa danos reais ao ambiente.

Considere facilitar as simulações cibernéticas em intervalos regulares. Cada tipo de simulação pode oferecer benefícios exclusivos aos participantes e à organização como um todo. Portanto, você pode optar por começar com tipos de simulação menos complexos (como simulações teóricas) e avançar para tipos de simulação mais complexos (simulações da equipe vermelha). Você deve selecionar um tipo de simulação com base em sua maturidade de segurança, recursos e resultados desejados. Alguns clientes podem não optar por realizar simulações da equipe vermelha devido à complexidade e ao custo.

Etapas da implementação

Independentemente do tipo de simulação que você escolher, as simulações geralmente seguem estas etapas de implementação:

1. Defina os principais elementos do exercício: defina o cenário e os objetivos da simulação. Ambos devem ter aceitação da liderança.
2. Identifique as principais partes interessadas: no mínimo, um exercício precisa de facilitadores e participantes. Dependendo do cenário, outras partes interessadas, como departamento jurídico, de comunicação ou liderança executiva, podem estar envolvidos.
3. Crie e teste o cenário: talvez o cenário precise ser redefinido durante a criação se elementos específicos não forem viáveis. Espera-se um cenário finalizado como resultado dessa etapa.
4. Facilite a simulação: o tipo de simulação determina a facilitação usada (um cenário impresso em comparação a um cenário simulado altamente técnico). Os facilitadores devem alinhar suas táticas de facilitação aos objetos da simulação e envolver todos os participantes sempre que possível para proporcionar o máximo benefício.
5. Desenvolva o relatório pós-ação (AAR): identifique as áreas que funcionaram bem, aquelas que podem ser melhoradas e possíveis déficits. O AAR deve medir a eficácia da simulação, bem como a resposta da equipe ao evento simulado, para que o progresso possa ser monitorado ao longo do tempo com simulações futuras.

Recursos

Documentos relacionados:

- [AWS Incident Response Guide](#) (Guia de resposta a incidentes da AWS)

Vídeos relacionados:

- [AWS GameDay - Security Edition](#) (Dia de jogo da AWS: edição de segurança)

Operações

As operações são a base da resposta a incidentes. É aqui que ocorrem as ações de resposta e atenuação de incidentes de segurança. As operações incluem estas cinco fases: detecção, análise, contenção, erradicação e recuperação. As descrições dessas fases e dos objetivos podem ser encontradas na tabela a seguir.

Fase	Objetivo
Detecção	Identifique um possível evento de segurança.
Análise	Determine se o evento de segurança é um incidente e avalie o escopo do incidente.
Contenção	Minimize e limite o escopo do evento de segurança.
Erradicação	Remova recursos ou artefatos não autorizados relacionados ao evento de segurança. Implemente atenuações que causaram o incidente de segurança.
Recuperação	Restaure os sistemas ao estado seguro conhecido e monitore esses sistemas para verificar se não há retorno da ameaça.

As fases devem servir como orientação quando você responde e atua em incidentes de segurança, a fim de responder de forma eficaz e robusta. As ações reais realizadas vão variar de acordo com o incidente. Um incidente envolvendo ransomware, por exemplo, terá um conjunto de etapas de resposta a serem seguidas diferente do que o de um incidente que envolva um bucket público do Amazon S3. Além disso, essas fases não acontecem necessariamente de modo sequencial. Após a contenção e a erradicação, talvez seja necessário retornar à análise para entender se suas ações foram eficazes.

A preparação completa de seu pessoal, processos e tecnologia é fundamental para ser eficaz nas operações. Dessa forma, siga as práticas recomendadas da seção [Preparação](#) para poder responder com eficácia a um evento de segurança ativo.

Para saber mais, consulte a seção [Operações](#) do Guia de resposta a incidentes de segurança da AWS.

Atividade pós-incidente

O cenário de ameaças está mudando constantemente, e é importante ser igualmente dinâmico na capacidade de sua organização de proteger seus ambientes com eficácia. A chave para a melhoria contínua é iterar os resultados de seus incidentes e simulações a fim de melhorar seus recursos para detectar, responder e investigar com eficácia possíveis incidentes de segurança, reduzindo suas possíveis vulnerabilidades, o tempo de resposta e o retorno às operações seguras. Os mecanismos a seguir podem ajudar você a verificar se sua organização continua preparada com os recursos e os conhecimentos mais recentes para responder com eficácia, independentemente da situação.

Práticas recomendadas

- [SEC10-BP08 Estabeleça uma estrutura para aprender com os incidentes](#)

SEC10-BP08 Estabeleça uma estrutura para aprender com os incidentes

A implementação de uma framework de lições aprendidas e da capacidade de análise da causa raiz não só ajudará a melhorar os recursos de resposta a incidentes, mas também ajudará a evitar que o incidente se repita. Ao aprender com cada incidente, você pode ajudar a evitar a repetição dos mesmos erros, exposições ou configurações incorretas, não apenas melhorando seu procedimento de segurança, mas também minimizando o tempo perdido em situações evitáveis.

Nível de risco exposto se esta prática recomendada não for estabelecida: Médio

Orientação para implementação

É importante implementar uma framework de lições aprendidas que estabeleça e alcance, em nível geral, os seguintes pontos:

- Quando as lições são aprendidas?
- O que está envolvido no processo de lições aprendidas?
- Como as lições aprendidas são colocadas em prática?
- Quem está envolvido no processo e como?
- Como as áreas de melhoria serão identificadas?
- Como você garantirá que as melhorias sejam monitoradas e implementadas de forma eficaz?

A estrutura não deve se concentrar em culpar os indivíduos, mas sim na melhoria de ferramentas e processos.

Etapas da implementação

Além dos resultados de alto nível listados acima, é importante garantir que você faça as perguntas certas para obter o máximo valor (informações que levem a melhorias práticas) do processo. Considere estas perguntas para ajudar você a começar a promover discussões sobre as lições aprendidas:

- Qual foi o incidente?
- Quando o incidente foi identificado pela primeira vez?
- Como ele foi identificado?
- Quais sistemas alertaram sobre a atividade?
- Quais sistemas, serviços e dados estavam envolvidos?
- O que ocorreu especificamente?
- O que funcionou bem?
- O que não funcionou bem?
- Quais processos ou procedimentos falharam ou não tiveram a escala ajustada para responder ao incidente?
- O que pode ser melhorado nas seguintes áreas:
 - Pessoas

- As pessoas que precisavam ser contatadas estavam realmente disponíveis e a lista de contatos estava atualizada?
- As pessoas estavam perdendo treinamentos ou não tinham os recursos necessários para responder e investigar o incidente com eficácia?
- Os recursos apropriados estavam prontos e disponíveis?
- Processo
 - Os processos e procedimentos foram seguidos?
 - Os processos e procedimentos foram documentados e estavam disponíveis para esse (tipo de) incidente?
 - Estavam faltando processos e procedimentos necessários?
 - Os respondentes conseguiram obter acesso oportuno às informações necessárias para responder ao problema?
- Tecnologia
 - Os sistemas de alerta existentes identificaram e alertaram efetivamente sobre a atividade?
 - Como poderíamos ter reduzido o tempo de detecção em 50%?
 - Os alertas existentes precisam ser aprimorados ou novos alertas precisam ser criados para esse (tipo de) incidente?
 - As ferramentas existentes permitiram uma investigação (pesquisa/análise) eficaz do incidente?
 - O que pode ser feito para ajudar a identificar esse (tipo de) incidente mais cedo?
 - O que pode ser feito para ajudar a evitar que esse (tipo de) incidente ocorra novamente?
 - Quem é o proprietário do plano de melhoria e como você testará se ele foi implementado?
 - Qual é o cronograma para que os controles e processos adicionais de monitoramento ou prevenção sejam implementados e testados?

Essa lista não inclui tudo, mas serve como ponto de partida para identificar quais são as necessidades da organização e da empresa e como você pode analisá-las para aprender com os incidentes de forma mais eficaz e melhorar constantemente seu procedimento de segurança. O mais importante é começar incorporando as lições aprendidas como parte padrão do processo de resposta a incidentes, da documentação e das expectativas das partes interessadas.

Recursos

Documentos relacionados:

- [AWS Security Incident Response Guide - Establish a framework for learning from incidents](#)
- [NCSC CAF guidance - Lessons learned \(Orientações do NCSC CAF: lições aprendidas\)](#)

Segurança de aplicações

A segurança de aplicações (AppSec) retrata o processo geral de como projetar, criar e testar as propriedades de segurança das workloads desenvolvidas. Você precisa treinar a equipe adequadamente em sua organização, entender as propriedades de segurança de sua infraestrutura de compilação e lançamento e utilizar a automação para identificar problemas de segurança.

Adotar testes de segurança de aplicações como parte regular do ciclo de vida de desenvolvimento de software (SDLC) e processos de pós-lançamento ajuda a garantir que você tenha um mecanismo estruturado para identificar, corrigir e impedir que problemas de segurança de aplicações entrem no ambiente de produção.

Sua metodologia de desenvolvimento de aplicações deve incluir controles de segurança à medida que você projeta, cria, implanta e opera suas workloads. Ao fazer isso, alinhe o processo para redução contínua de defeitos e redução da dívida técnica. Por exemplo, o uso de modelagem de ameaças na fase de design ajuda a detectar falhas de design precocemente, o que torna mais fácil e menos caro corrigi-las em contraposição a aguardar e mitigá-las posteriormente.

O custo e a complexidade para resolver defeitos geralmente serão menores quanto mais no princípio você estiver no SDLC. A forma mais fácil de resolver problemas é não os ter. Por isso, começar com um modelo de ameaças ajuda você a se concentrar nos resultados corretos da fase de design. À medida que seu programa de AppSec amadurece, é possível aumentar a quantidade de testes realizados usando automação, aumentar a fidelidade do feedback para os criadores e reduzir o tempo necessário para as avaliações de segurança. Todas essas ações melhoram a qualidade do software desenvolvido e aumentam a velocidade de entrega de recursos à produção.

Essas diretrizes de implementação concentram-se em quatro áreas: organização e cultura, segurança do pipeline, segurança no pipeline e gerenciamento de dependências. Cada área oferece um conjunto de princípios que você pode implementar, bem como uma visão completa de como projetar, desenvolver, criar, implantar e operar workloads.

Na AWS, há várias abordagens para lidar com seu programa de segurança de aplicações. Algumas dessas abordagens dependem de tecnologia, enquanto outras se concentram na equipe e em aspectos organizacionais do programa de segurança de aplicações.

Práticas recomendadas

- [SEC11-BP01 Treinar para segurança de aplicações](#)
- [SEC11-BP02 Automatizar o teste durante o ciclo de vida de desenvolvimento e lançamento](#)

- [SEC11-BP03 Realizar teste de penetração regular](#)
- [SEC11-BP04 Análises manuais de código](#)
- [SEC11-BP05 Centralizar serviços para pacotes e dependências](#)
- [SEC11-BP06 Implantar software programaticamente](#)
- [SEC11-BP07 Avaliar regularmente as propriedades de segurança dos pipelines](#)
- [SEC11-BP08 Criar um programa que incorpore a propriedade de segurança nas equipes de workload](#)

SEC11-BP01 Treinar para segurança de aplicações

Forneça treinamento aos criadores em sua organização sobre práticas comuns para promover a segurança no desenvolvimento e na operação de aplicações. A adoção de práticas de desenvolvimento com foco na segurança ajuda a diminuir a probabilidade de problemas que são detectados somente no estágio de avaliação da segurança.

Resultado desejado: o software deve ser projetado e criado com a segurança em mente. Quando os criadores em uma organização são treinados em práticas de desenvolvimento seguras que começam com um modelo de ameaças, isso melhora a qualidade e a segurança gerais do software produzido. Essa abordagem pode reduzir o tempo de entrega do software ou de recursos porque não é necessário tanto retrabalho após o estágio de avaliação da segurança.

Para as finalidades desta prática recomendada, desenvolvimento seguro refere-se ao software que está sendo criado e às ferramentas ou aos sistemas compatíveis com o ciclo de vida de desenvolvimento de software (SDLC).

Antipadrões comuns:

- Aguardar uma avaliação da segurança e, depois, considerar as propriedades de segurança de um sistema.
- Deixar todas as decisões de segurança para a equipe de segurança.
- Não comunicar como as decisões tomadas no SDLC se relacionam às expectativas ou as políticas de segurança gerais da organização.
- Iniciar o processo de avaliação da segurança muito tardiamente.

Benefícios do estabelecimento desta prática recomendada:

- Melhor conhecimento dos requisitos organizacionais para a segurança na fase inicial do ciclo de desenvolvimento.
- Ser capaz de identificar e solucionar possíveis problemas de segurança com maior rapidez, promovendo uma entrega de recursos mais rápida.
- Maior qualidade do software e dos sistemas.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

Ofereça treinamento aos criadores em sua organização. Iniciar um curso sobre [modelagem de ameaças](#) é uma boa base para ajudar a treinar para segurança. Preferencialmente, os criadores devem ser capazes de acessar de forma independente as informações relevantes às respectivas workloads. Esse acesso os ajuda a tomar decisões embasadas sobre as propriedades de segurança dos sistemas criados por eles sem a necessidade de solicitar outra equipe. O processo para envolver a equipe de segurança para avaliações deve ser claramente definido e simples de seguir. As etapas do processo de avaliação devem ser incluídas no treinamento de segurança. Quando houver padrões ou modelos de implementação disponíveis, eles deverão ser simples de encontrar e vincular aos requisitos de segurança gerais. Considere usar o [AWS CloudFormation](#), as [estruturas do AWS Cloud Development Kit \(AWS CDK\)](#), o [Service Catalog](#) ou outras ferramentas de modelo para reduzir a necessidade de configuração personalizada.

Etapas da implementação

- Oferecer aos criadores um curso sobre [modelagem de ameaças](#) para criar uma boa base e ajudar a treiná-los a pensar em segurança.
- Conceder acesso ao treinamento do [Treinamento da AWS and Certification](#), do setor ou de parceiros da AWS.
- Fornecer treinamento sobre o processo de avaliação da segurança de sua organização, que esclarece a divisão de responsabilidades entre a equipe de segurança, as equipes de workload e outras partes interessadas.
- Publicar orientações de autoatendimento sobre como atender aos seus requisitos de segurança, inclusive códigos de exemplo e modelos, se disponíveis.
- Obter feedback regularmente de equipes de criadores sobre a experiência deles com o processo e o treinamento de processo de avaliação da segurança e usar esse feedback para promover melhorias.

- Utilizar dias de jogo ou campanhas de bug bash para ajudar a reduzir o número de problemas e aumentar as habilidades de seus criadores.

Recursos

Práticas recomendadas relacionadas:

- [SEC11-BP08 Criar um programa que incorpore a propriedade de segurança nas equipes de workload](#)

Documentos relacionados:

- [Treinamento da AWS and Certification](#)
- [Como pensar sobre governança de segurança na nuvem](#)
- [Como abordar a modelagem de ameaças](#)
- [Como acelerar o treinamento: o AWS Skills Guild](#)

Vídeos relacionados:

- [Segurança proativa: considerações e abordagens](#)

Exemplos relacionados:

- [Workshop sobre modelagem de ameaças](#)
- [Conscientização do setor para desenvolvedores](#)

Serviços relacionados:

- [AWS CloudFormation](#)
- [Estruturas do AWS Cloud Development Kit \(AWS CDK\) \(AWS CDK\)](#)
- [Service Catalog](#)
- [AWS BugBust](#)

SEC11-BP02 Automatizar o teste durante o ciclo de vida de desenvolvimento e lançamento

Automatize o teste das propriedades de segurança durante o ciclo de vida de desenvolvimento e lançamento. Com a automação, é mais fácil identificar de forma consistente e repetível possíveis problemas no software antes do lançamento, o que reduz o risco de problemas de segurança no software que está sendo fornecido.

Resultado esperado: o objetivo do teste automatizado é oferecer uma forma programática de detectar possíveis problemas precocemente e com frequência ao longo do ciclo de vida de desenvolvimento. Ao automatizar o teste de regressão, você pode executar novamente testes funcionais e não funcionais para verificar se o software testado anteriormente ainda funciona da forma esperada após uma alteração. Ao definir testes de unidade de segurança para conferir configurações incorretas comuns, como uma autenticação ausente ou danificada, é possível identificar e resolver esses problemas logo no início do processo de desenvolvimento.

A automação de testes utiliza casos de teste para um propósito específico para validação de aplicações, com base nos requisitos e na funcionalidade desejada da aplicação. O resultado dos testes automatizados baseia-se na comparação da saída do teste gerado com a respectiva saída esperada, o que acelera o ciclo de vida dos testes em geral. As metodologias de teste, como teste de regressão e pacotes de teste de unidade, são mais adequadas para automação. A automação dos testes de propriedades de segurança possibilita aos criadores receber feedback automatizado sem precisar esperar por uma avaliação da segurança. Os testes automatizados em forma de análise de código estático ou dinâmico podem melhorar a qualidade do código e ajudar a detectar possíveis problemas de software no ciclo de vida de desenvolvimento.

Antipadrões comuns:

- Não comunicar os casos de teste e os resultados dos testes automatizados.
- Realizar os testes automatizados somente antes de um lançamento.
- Automatizar casos de teste com requisitos que mudam com frequência.
- Não fornecer orientações sobre como abordar os resultados dos testes de segurança.

Benefícios do estabelecimento desta prática recomendada:

- Redução da dependência de pessoas que avaliam as propriedades de segurança dos sistemas.

- Descobertas consistentes em vários fluxos de trabalho que melhoram a consistência.
- Redução da probabilidade de introduzir problemas de segurança no software de produção.
- Redução do período de tempo entre a detecção e a correção devido à detecção mais antecipada de problemas de software.
- Maior visibilidade do problema sistêmico ou repetido entre os vários fluxos de trabalho, o que pode ser utilizado para promover melhorias em toda a organização.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientação de implementação

Ao criar um software, adote vários mecanismos de teste para garantir que você esteja testando os requisitos funcionais da aplicação, com base na respectiva lógica de negócios e em requisitos não funcionais, os quais se concentram na confiabilidade, performance e segurança da aplicação.

O teste de segurança de aplicação estática (SAST) analisa padrões de segurança anômalos no código-fonte e fornece indicações de código propenso a defeitos. O SAST depende de entradas estáticas, como documentação (especificação de requisitos, documentação e especificações de design) e código-fonte da aplicação, para testar uma série de problemas de segurança conhecidos. Os analisadores de código estático podem ajudar a acelerar a análise de grandes volumes de código. O [NIST Quality Group](#) oferece uma comparação de [analisadores de segurança de código-fonte](#), o que inclui ferramentas de código aberto para [leitores de código de byte](#) e [leitores de código binário](#).

Complemente seu teste estático com metodologias de teste de segurança de análise dinâmica (DAST), que realizam testes na aplicação em execução a fim de identificar comportamento possivelmente inesperado. O teste dinâmico pode ser utilizado para detectar possíveis problemas que não são detectáveis por meio de análise estática. Por meio dos testes nos estágios de repositório de código, compilação e pipeline, é possível impedir que diferentes tipos de problema em potencial ocorram no código. O [Amazon CodeWhisperer](#) oferece recomendações de código, como verificação de segurança, no IDE do criador. O [Amazon CodeGuru Reviewer](#) pode identificar problemas críticos, problemas de segurança e bugs difíceis de detectar durante o desenvolvimento da aplicação e oferece recomendações para melhorar a qualidade do código.

O workshop [Segurança para desenvolvedores](#) utiliza ferramentas de desenvolvedor da AWS, como [AWS CodeBuild](#), [AWS CodeCommit](#) e [AWS CodePipeline](#), para automação de pipeline de lançamento que inclui as metodologias de teste SAST e DAST.

À medida que você avança no SDLC, estabeleça um processo iterativo que inclua avaliações de aplicação periódicas com sua equipe de segurança. O feedback coletado dessas avaliações de segurança deve ser abordado e validado como parte de sua avaliação de prontidão do lançamento. Essas avaliações estabelecem um procedimento de segurança robusto de aplicações e fornecem aos criadores feedback útil para resolver possíveis problemas.

Etapas da implementação

- Implementar um IDE consistente, análise de código e ferramentas de CI/CD que incluam teste de segurança.
- Considerar quando no SDLC é adequado bloquear pipelines em vez de apenas notificar os criadores de que problemas precisam ser corrigidos.
- O workshop [Segurança para desenvolvedores](#) fornece um exemplo de como integrar testes estáticos e dinâmicos a um pipeline de lançamento.
- Realizar testes ou análise de código com ferramentas automatizadas, como o [Amazon CodeWhisperer](#) integrado a IDEs de desenvolvedores e o [Amazon CodeGuru Reviewer](#) para verificação do código na confirmação, ajuda os criadores a obter feedback no momento certo.
- Ao criar com o AWS Lambda, é possível usar o [Amazon Inspector](#) para verificar o código de aplicação em suas funções.
- O workshop [CI/CD na AWS](#) fornece um ponto de partida para criar pipelines de CI/CD na AWS.
- Quando testes automatizados são incluídos em pipelines de CI/CD, você precisa usar um sistema de emissão de tíquetes para rastrear a notificação e a correção de problemas de software.
- Para testes de segurança que podem gerar descobertas, a vinculação com orientações para correção ajuda os criadores a melhorar a qualidade do código.
- Analise regularmente as descobertas das ferramentas automatizadas para priorizar a próxima automação, o treinamento de criadores ou a campanha de conscientização.

Recursos

Documentos relacionados:

- [Entrega contínua e implantação contínua](#)
- [Parceiros com competência em DevOps da AWS](#)
- [Parceiros de competência em segurança da AWS](#) para segurança da aplicação

- [Como escolher uma abordagem de CI/CD do Well-Architected](#)
- [Monitorar eventos do CodeCommit no Amazon EventBridge e no Amazon CloudWatch Events](#)
- [Análise da detecção de segredos no Amazon CodeGuru Review](#)
- [Acelerar implantações na AWS com governança efetiva](#)
- [Como a AWS aborda a automação de implantações seguras e sem intervenção manual](#)

Vídeos relacionados:

- [Sem intervenção manual: como automatizar os pipelines de entrega contínua na Amazon](#)
- [Como automatizar pipelines CI/CD entre contas](#)

Exemplos relacionados:

- [Conscientização do setor para desenvolvedores](#)
- [Governança do AWS CodePipeline](#)
- Workshop [Segurança para desenvolvedores](#)
- [Workshop sobre CI/CD da AWS](#)

SEC11-BP03 Realizar teste de penetração regular

Realize teste de penetração regular do software. Esse mecanismo ajuda a identificar possíveis problemas de software que não podem ser detectados pelo teste automatizado ou por uma análise manual do código. Ele também ajuda você a entender a eficácia dos controles de detecção. O teste de penetração deve tentar determinar se o software pode ser executado de formas inesperadas; por exemplo, expondo dados que devem ser protegidos ou concedendo permissões mais amplas que o esperado.

Resultado desejado: o teste de penetração é usado para detectar, corrigir e validar as propriedades de segurança da aplicação. O teste de penetração regular e programado deve ser realizado como parte do ciclo de vida de desenvolvimento de software (SDLC). As descobertas do teste de penetração devem ser abordadas antes do lançamento do software. Você precisa analisar as descobertas do teste de penetração para identificar se há problemas que podem ser encontrados usando a automação. Ter um processo de teste de penetração regular e repetível que inclua um

mecanismo de feedback ativo ajuda a transmitir as orientações aos criadores e melhora a qualidade do software.

Antipadrões comuns:

- Realizar um teste de penetração somente para problemas de segurança conhecidos ou prevalentes.
- Realizar um teste de penetração em aplicações sem ferramentas e bibliotecas de terceiro dependentes.
- Realizar um teste de penetração em aplicações em busca de problemas de segurança de pacote e não avaliar a lógica de negócios implementada.

Benefícios do estabelecimento desta prática recomendada:

- Maior confiança nas propriedades de segurança do software antes do lançamento.
- Oportunidade de identificar padrões de aplicação preferenciais, o que aumenta a qualidade do software.
- Um ciclo de feedback que identifica mais cedo no ciclo de desenvolvimento quando a automação ou treinamento adicional pode melhorar as propriedades de segurança do software.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientação de implementação

O teste de penetração é um exercício de teste de segurança estruturado em que você executa cenários de violação de segurança planejados a fim de detectar, corrigir e validar controles de segurança. Os testes de penetração começam com o reconhecimento, durante o qual os dados são coletados com base no design atual da aplicação e nas respectivas dependências. Uma lista selecionada de cenários de teste específicos de segurança é criada e executada. A principal finalidade desses testes é revelar problemas de segurança em sua aplicação, que podem ser explorados para obter acesso não intencional ao seu ambiente ou acesso não autorizado aos dados. Você precisa realizar o teste de penetração ao lançar novos recursos ou sempre que sua aplicação passar por alterações importantes na implementação técnica ou de funções.

É necessário identificar o estágio mais apropriado do ciclo de vida de desenvolvimento para realizar o teste de penetração. Esse teste deve ocorrer em uma fase tardia o suficiente para que

a funcionalidade do sistema esteja próxima ao estado de lançamento pretendido, mas com tempo suficiente para corrigir todos os problemas.

Etapas da implementação

- Ter um processo estruturado sobre como definir o escopo do teste de penetração. Basear esse processo no [modelo de ameaças](#) é uma boa forma de manter o contexto.
- Identificar o estágio apropriado do ciclo de vida de desenvolvimento para realizar o teste de penetração, que deve ser quando houver o mínimo de alterações esperadas na aplicação e houver tempo suficiente para realizar a correção.
- Treinar os criadores sobre o que esperar das descobertas do teste de penetração e como ter informações sobre correção.
- Utilizar ferramentas para acelerar o processo de testes de penetração automatizando testes comuns ou repetíveis.
- Analisar as descobertas do teste de penetração para identificar problemas de segurança sistêmicos e utilizar esses dados para embasar testes automatizados adicionais e a instrução contínua dos criadores.

Recursos

Práticas recomendadas relacionadas:

- [SEC11-BP01 Treinar para segurança de aplicações](#)
- [SEC11-BP02 Automatizar o teste durante o ciclo de vida de desenvolvimento e lançamento](#)

Documentos relacionados:

- [O teste de penetração da AWS](#) fornece orientações detalhadas para teste de penetração na AWS
- [Acelerar implantações na AWS com governança efetiva](#)
- [Parceiros de competência em segurança da AWS](#)
- [Modernize sua arquitetura de teste de penetração no AWS Fargate](#)
- [AWS Fault Injection Simulator](#)

Exemplos relacionados:

- [Como automatizar testes de API com o AWS CodePipeline](#) (GitHub)
- [Assistente de segurança automatizado](#) (GitHub)

SEC11-BP04 Análises manuais de código

Realize uma análise manual do código do software que você produz. Esse processo ajuda a verificar se a pessoa que escreveu o código não é a única que está conferindo a qualidade dele.

Resultado desejado: a inclusão de uma etapa de análise de código manual durante o desenvolvimento melhora a qualidade do software que está sendo criado, ajuda a melhorar as habilidades de membros menos experientes da equipe e oferece uma oportunidade de identificar locais onde a automação pode ser usada. É possível oferecer compatibilidade com as análises de código manuais com ferramentas e testes automatizados.

Antipadrões comuns:

- Não realizar análises de código antes da implantação.
- Ter a mesma pessoa para escrever e analisar o código.
- Não utilizar a automação para auxiliar ou orquestrar as análises de código.
- Não treinar os criadores em segurança de aplicações antes de analisarem o código.

Benefícios do estabelecimento desta prática recomendada:

- Código de melhor qualidade.
- Maior consistência do desenvolvimento do código por meio da reutilização de abordagens comuns.
- Redução no número de problemas descobertos durante o teste de penetração e em estágios posteriores.
- Maior transferência de conhecimentos na equipe.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientações para a implementação

A etapa de análise deve ser implementada como parte do fluxo de gerenciamento de código geral. Os detalhes dependem da abordagem utilizada para ramificação, solicitações de pull e mesclagem.

Você pode utilizar o AWS CodeCommit ou soluções de terceiros, como GitHub, GitLab ou Bitbucket. Seja qual for o método utilizado, é importante verificar se seus processos precisam de análise de código antes da implantação em um ambiente de produção. O uso de ferramentas, como o [Amazon CodeGuru Reviewer](#), pode facilitar a orquestração do processo de análise do código.

Etapas da implementação

- Implementar uma etapa de análise manual como parte do fluxo de gerenciamento de código e realizar essa análise antes de prosseguir.
- Considerar o [Amazon CodeGuru Reviewer](#) para gerenciar e auxiliar nas análises de código.
- Implementar um fluxo de aprovação que exija a realização de uma análise de código antes de avançá-lo para o próximo estágio.
- Verificar se há um processo para identificar problemas encontrados durante as análises de código manuais que possam ser detectados automaticamente.
- Integrar a etapa de análise de código manual de forma que se alinhe às suas práticas de desenvolvimento de código.

Recursos

Práticas recomendadas relacionadas:

- [SEC11-BP02 Automatizar o teste durante o ciclo de vida de desenvolvimento e lançamento](#)

Documentos relacionados:

- [Trabalhar com solicitações de pull em repositórios do AWS CodeCommit](#)
- [Trabalhar com modelos de regra de aprovação no AWS CodeCommit](#)
- [Sobre solicitações de pull no GitHub](#)
- [Análises de código automatizadas com o Amazon CodeGuru Reviewer](#)
- [Automatizar a detecção de vulnerabilidades de segurança e bugs em pipelines de CI/CD com o uso da CLI do Amazon CodeGuru Reviewer](#)

Vídeos relacionados:

- [Melhoria contínua da qualidade do código com o Amazon CodeGuru](#)

Exemplos relacionados:

- Workshop [Segurança para desenvolvedores](#)

SEC11-BP05 Centralizar serviços para pacotes e dependências

Forneça serviços centralizados a equipes de criadores para obter pacotes de software e outras dependências. Isso permite a validação de pacotes antes que eles sejam incluídos no software que você escreve e fornece uma fonte de dados para a análise do software que está sendo usado na sua organização.

Resultado desejado: o software é composto de um conjunto de outros pacotes de software além do código que está sendo escrito. Isso simplifica o consumo de implementações de funcionalidades que são utilizadas repetidamente, como um analisador JSON ou uma biblioteca de criptografia. A centralização lógica das fontes desses pacotes e dependências oferece um mecanismo para as equipes de segurança validarem as propriedades dos pacotes antes de eles serem utilizados. Essa abordagem também reduz o risco de um problema inesperado ser provocado por uma alteração em um pacote existente ou pela inclusão de pacotes arbitrários diretamente da Internet pelas equipes de criadores. Utilize essa abordagem em conjunto com os fluxos de testes manuais e automatizados para aumentar a confiança na qualidade do software que está sendo desenvolvido.

Antipadrões comuns:

- Extrair pacotes de repositórios arbitrários na Internet.
- Não testar novos pacotes antes de disponibilizá-los aos criadores.

Benefícios do estabelecimento desta prática recomendada:

- Melhor entendimento de quais pacotes estão sendo utilizados no software que está sendo criado.
- Capacidade de notificar as equipes de workload quando um pacote precisa ser atualizado com base no entendimento de quem está usando o quê.
- Redução do risco de um pacote com problemas ser incluído em seu software.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: médio

Orientação de implementação

Forneça serviços centralizados para pacotes e dependências de uma forma simples para os criadores consumirem. Serviços centralizados podem ser centralizados logicamente em vez de implementados como um sistema monolítico. Essa abordagem possibilita fornecer serviços de uma forma que atenda às necessidades dos criadores. Você precisa implementar uma forma eficiente de adicionar pacotes ao repositório quando ocorrem atualizações ou surgem novos requisitos. Serviços da AWS como o [AWS CodeArtifact](#) ou soluções semelhantes de parceiros da AWS oferecem uma forma de entregar esse recurso.

Etapas da implementação:

- Implementar um serviço de repositório centralizado logicamente disponível em todos os ambientes onde o software é desenvolvido.
- Incluir acesso ao repositório como parte do processo de provisionamento de Conta da AWS.
- Criar automação para testar pacotes antes de serem publicados em um repositório.
- Manter métricas dos pacotes mais utilizados, das linguagens e das equipes com a maior quantidade de alterações.
- Fornecer um mecanismo automatizado para as equipes de criadores solicitarem novos pacotes e fornecerem feedback.
- Verificar regularmente os pacotes em seu repositório para identificar o possível impacto de problemas recém-descobertos.

Recursos

Práticas recomendadas relacionadas:

- [SEC11-BP02 Automatizar o teste durante o ciclo de vida de desenvolvimento e lançamento](#)

Documentos relacionados:

- [Acelerar implantações na AWS com governança efetiva](#)
- [Aumentar a segurança de seu pacote com o kit de ferramentas CodeArtifact Package Origin Control](#)
- [Detectar problemas de segurança no registro em log com o Amazon CodeGuru Reviewer](#)
- [Níveis de cadeia de suprimentos para artefatos de software \(SLSA\)](#)

Vídeos relacionados:

- [Segurança proativa: considerações e abordagens](#)
- [A filosofia de segurança da AWS \(re:Invent 2017\)](#)
- [Quando a segurança, a proteção e a urgência importam: lidar com o Log4Shell](#)

Exemplos relacionados:

- [Pipeline de publicação de pacotes de várias regiões \(GitHub\)](#)
- [Publicar módulos Node.js no AWS CodeArtifact usando o AWS CodePipeline \(GitHub\)](#)
- [Exemplo de pipeline do AWS CDK Java CodeArtifact \(GitHub\)](#)
- [Distribuir pacotes privados do .NET NuGet com o AWS CodeArtifact \(GitHub\)](#)

SEC11-BP06 Implantar software programaticamente

Faça implantações de software de forma programática quando possível. Essa abordagem diminui a probabilidade de falha em uma implantação ou da introdução de um problema inesperado devido a erro humano.

Resultado desejado: manter as pessoas longe dos dados é um princípio essencial da criação segura na Nuvem AWS. Esse princípio inclui como implantar seu software.

Os benefícios de não contar com pessoas para implantar software é a maior confiança de que o componente testado é o que será implantado e de que a implantação sempre é realizada de forma consistente. O software não deve precisar de alterações para funcionar em diferentes ambientes. O uso dos princípios de desenvolvimento de aplicações de 12 fatores, especificamente a externalização da configuração, possibilita implantar o mesmo código em vários ambientes sem a necessidade de alterações. Assinar de forma criptográfica os pacotes de software é uma boa maneira de garantir que nada tenha sido alterado entre os ambientes. O resultado geral dessa abordagem é reduzir o risco em seu processo de alterações e melhorar a consistência das versões do software.

Antipadrões comuns:

- Implantar software manualmente em produção.
- Realizar alterações manualmente no software para suprir diferentes ambientes.

Benefícios do estabelecimento desta prática recomendada:

- Maior confiança no processo de lançamento de software.
- Redução do risco de uma alteração com falha afetar a funcionalidade dos negócios.
- Maior cadência de lançamentos devido ao menor risco de alterações.
- Recurso de reversão automática para eventos inesperados durante a implantação.
- Capacidade de comprovar de forma criptográfica que o software testado é o software implantado.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientações para a implementação

Crie a infraestrutura de sua Conta da AWS para remover o acesso humano persistente dos ambientes e use ferramentas de CI/CD para realizar implantações. Projete suas aplicações de forma que os dados da configuração específica do ambiente sejam obtidos de uma fonte externa, como o [AWS Systems Manager Parameter Store](#). Assine pacotes depois de testados e valide essas assinaturas durante a implantação. Configure seus pipelines de CI/CD para enviar código da aplicação e usar canários para confirmar a implantação bem-sucedida. Utilize ferramentas como o [AWS CloudFormation](#) ou o [AWS CDK](#) para definir sua infraestrutura; depois, use o [AWS CodeBuild](#) e o [AWS CodePipeline](#) para realizar operações de CI/CD.

Etapas da implementação

- Criar pipelines de CI/CD bem definidos para simplificar o processo de implantação.
- O uso do [AWS CodeBuild](#) e do [AWS Code Pipeline](#) para oferecer recurso de CI/CD simplifica a integração de teste de segurança aos seus pipelines.
- Seguir as orientações sobre separação de ambientes no whitepaper [Organizar seu ambiente da AWS com o uso de várias contas](#).
- Garantir que não haja nenhum acesso humano persistente aos ambientes nos quais as workloads de produção estão em execução.
- Projetar as aplicações para oferecer compatibilidade com a externalização de dados de configuração.
- Considerar a implantação com o uso do modelo de implantação azul/verde.
- Implementar canários para validar a implantação bem-sucedida do software.
- Utilizar ferramentas criptográficas, como o [AWS Signer](#) ou o [AWS Key Management Service \(AWS KMS\)](#), para assinar e confirmar os pacotes de software que você está implantando.

Recursos

Práticas recomendadas relacionadas:

- [SEC11-BP02 Automatizar o teste durante o ciclo de vida de desenvolvimento e lançamento](#)

Documentos relacionados:

- [Workshop sobre CI/CD da AWS](#)
- [Acelerar implantações na AWS com governança efetiva](#)
- [Automatizar uma implantação prática e sem intervenção manual](#)
- [Assinatura de código com o uso de CA privada do AWS Certificate Manager e chaves assimétricas do AWS Key Management Service\)](#)
- [Assinatura de código: um controle de integridade e confiança para o AWS Lambda](#)

Vídeos relacionados:

- [Sem intervenção manual: como automatizar os pipelines de entrega contínua na Amazon](#)

Exemplos relacionados:

- [Implantações azul/verde com o AWS Fargate](#)

SEC11-BP07 Avaliar regularmente as propriedades de segurança dos pipelines

Aplicar os princípios do pilar Segurança do Well-Architected aos seus pipelines, com atenção especial à separação das permissões. Avalie as propriedades de segurança de sua infraestrutura de pipelines. O gerenciamento eficaz da segurança dos pipelines permite que você forneça segurança ao software que passa pelos pipelines.

Resultado desejado: os pipelines utilizados para criar e implantar o software devem seguir as mesmas práticas recomendadas que qualquer outra workload em seu ambiente. Os testes implementados nos pipelines não devem ser editáveis pelos criadores que os estão utilizando. Os pipelines só devem ter as permissões necessárias para as implantações que eles estão realizando e devem implementar proteções para evitar a implantação em ambientes errados. Os pipelines não

devem contar com credenciais de longo prazo e devem ser configurados para emitir o estado de forma que a integridade dos ambientes de compilação possa ser validada.

Antipadrões comuns:

- Testes de segurança que podem ser ignorados pelos criadores.
- Permissões excessivamente amplas para pipelines de implantação.
- Pipelines não configurados para validar entradas.
- Ausência de análise regular das permissões associadas à infraestrutura de CI/CD.
- Uso de credenciais de longo prazo ou codificadas.

Benefícios do estabelecimento desta prática recomendada:

- Maior confiança na integridade do software que está sendo criado e implantado pelos pipelines.
- Capacidade de interromper uma implantação quando há atividade suspeita.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: alto

Orientação de implementação

Iniciar com serviços de CI/CD gerenciados que ofereçam compatibilidade com perfis do IAM reduz o risco de vazamento de credenciais. Aplicar os princípios do pilar Segurança à sua infraestrutura de pipeline de CI/CD pode ajudar você a determinar onde é possível realizar melhorias de segurança. Seguir a [Arquitetura de referência de pipelines de implantação da AWS](#) é um bom ponto de partida para criar seus ambientes de CI/CD. Analisar regularmente a implementação de pipelines e analisar comportamentos inesperados nos logs pode ajudar você a entender os padrões de uso dos pipelines que estão sendo utilizados para implantar o software.

Etapas da implementação

- Iniciar com a [Arquitetura de referência de pipeline de implantação da AWS](#).
- Considerar o uso do [AWS IAM Access Analyzer](#) para gerar de forma programática as políticas de privilégio mínimo do IAM para os pipelines.
- Integrar seus pipelines ao monitoramento e aos alertas de forma que você seja notificado de atividade inesperada ou anormal. Para serviços gerenciados da AWS, o [Amazon EventBridge](#) possibilita rotear dados para destinos, como o [AWS Lambda](#) ou o [Amazon Simple Notification Service](#) (Amazon SNS).

Recursos

Documentos relacionados:

- [Arquitetura de referência de pipeline de implantação da AWS](#)
- [Monitorar o AWS CodePipeline](#)
- [Práticas recomendadas de segurança para o AWS CodePipeline](#)

Exemplos relacionados:

- [Painel de monitoramento de DevOps](#) (GitHub)

SEC11-BP08 Criar um programa que incorpore a propriedade de segurança nas equipes de workload

Crie um programa ou mecanismo que capacite as equipes de criadores a tomar decisões de segurança sobre o software que elas estão criando. Ainda assim é necessário que sua equipe de segurança valide essas decisões durante uma avaliação, mas a incorporação da propriedade de segurança nas equipes de criadores aumenta a velocidade e segurança do processo de criação de workloads. Esse mecanismo também promove uma cultura de propriedade que afeta de forma positiva a operação dos sistemas que você cria.

Resultado desejado: para incorporar a propriedade de segurança e a tomada de decisão às equipes de criadores, você pode treinar os criadores a pensar sobre segurança ou incrementar o treinamento deles com pessoal de segurança incorporado ou associado às equipes de criadores. As duas abordagens são válidas e possibilitam à equipe tomar decisões de segurança de melhor qualidade logo no início do ciclo de desenvolvimento. Esse modelo de propriedade é baseado em treinamento para segurança de aplicações. Iniciar com o modelo de ameaças para a workload específica ajuda a direcionar o design thinking (pensamento de design) para o contexto apropriado. Outro benefício de ter uma comunidade de criadores concentrados em segurança ou um grupo de engenheiros de segurança que trabalhem com equipes de criadores é que você pode entender mais profundamente como o software é escrito. Esse entendimento ajuda você a determinar as próximas áreas de melhoria em seu recurso de automação.

Antipadrões comuns:

- Deixar todas as decisões de design de segurança para a equipe de segurança.
- Não abordar os requisitos de segurança cedo o suficiente no processo de desenvolvimento.
- Não obter feedback dos criadores e do pessoal de segurança sobre a operação do programa.

Benefícios do estabelecimento desta prática recomendada:

- Redução do tempo para concluir as avaliações de segurança.
- Redução dos problemas de segurança que são detectados apenas no estágio de avaliação da segurança.
- Melhoria da qualidade geral do software que está sendo escrito.
- Oportunidade de identificar e entender problemas sistêmicos ou áreas de melhoria de alto valor.
- Redução da quantidade de revisão necessária devido às descobertas da avaliação da segurança.
- Melhoria da percepção da função de segurança.

Nível de exposição a riscos se esta prática recomendada não for estabelecida: baixo

Orientações para a implementação

Comece com as orientações em [SEC11-BP01 Treinar para segurança de aplicações](#). Depois, identifique o modelo operacional para o programa que você acredita ser o melhor para a sua organização. Os dois padrões principais são treinar os criadores ou incorporar o pessoal de segurança às equipes de criadores. Depois de decidir sobre a abordagem inicial, você precisa criar um piloto com uma equipe de workload ou um grupo pequeno de equipes de workload para comprovar que o modelo funciona para sua organização. O apoio de liderança dos criadores e da segurança da organização contribui para a entrega e o sucesso do programa. À medida que você criar esse programa, é importante selecionar as métricas que podem ser utilizadas para mostrar o valor dele. Saber como a AWS resolveu esse problema é uma boa experiência de aprendizado. A prática recomendada é muito concentrada na mudança e cultura organizacionais. As ferramentas que você utiliza devem ser compatíveis com a colaboração entre as comunidades de criadores e de segurança.

Etapas da implementação

- Começar com o treinamento dos criadores para segurança de aplicações.
- Criar uma comunidade e um programa de integração para instruir os criadores.

- Selecionar um nome para o programa. Guardiões, patrocinadores ou defensores são utilizados com frequência.
- Identificar o modelo a ser utilizado: treinar criadores, incorporar engenheiros de segurança e ter perfis de segurança de afinidade.
- Identificar patrocinadores do projeto em grupos de segurança e de criadores e possivelmente em outros grupos relevantes.
- Rastrear as métricas do número de pessoas envolvidas no programa, o tempo gasto em avaliações e o feedback dos criadores e do pessoal de segurança. Utilizar essas métricas para realizar melhorias.

Recursos

Práticas recomendadas relacionadas:

- [SEC11-BP01 Treinar para segurança de aplicações](#)
- [SEC11-BP02 Automatizar o teste durante o ciclo de vida de desenvolvimento e lançamento](#)

Documentos relacionados:

- [Como abordar a modelagem de ameaças](#)
- [Como pensar sobre governança de segurança na nuvem](#)

Vídeos relacionados:

- [Segurança proativa: considerações e abordagens](#)

Conclusão

A segurança é um esforço contínuo. Quando ocorrem incidentes, eles devem ser tratados como oportunidades de melhorar a segurança da arquitetura. Ter controles fortes de identidade, automatizar respostas a eventos de segurança, proteger a infraestrutura em vários níveis e usar criptografia para gerenciar dados bem classificados proporcionam a defesa profunda que todas as empresas devem implementar. Esse trabalho é facilitado graças às funções programáticas e aos recursos e serviços da AWS discutidos neste documento.

A AWS se esforça para ajudá-lo a criar e operar arquiteturas que protegem informações, sistemas e ativos, enquanto agregam valor de negócios.

Colaboradores

Os indivíduos e empresas a seguir contribuíram para este documento:

- Sarita Dharankar, líder do pilar Segurança do Well-Architected, Amazon Web Services
- Adam Cerini, arquiteto sênior de soluções, Amazon Web Services
- Bill Shinn, diretor sênior no escritório do CISO, Amazon Web Services
- Brigid Johnson, gerente sênior de desenvolvimento de software, AWS Identity, Amazon Web Services
- Byron Pogson, arquiteto sênior de soluções, Amazon Web Services
- Charlie Hammell, arquiteto-chefe empresarial, Amazon Web Services
- Darran Boyd, arquiteto principal de soluções de segurança, serviços financeiros, Amazon Web Services
- Dave Walker, arquiteto de soluções, segurança e conformidade, Amazon Web Services
- John Formento, arquiteto sênior de soluções, Amazon Web Services
- Paul Hawkins, diretor sênior no Escritório do CISO, Amazon Web Services
- Sam Elmalak, líder sênior de tecnologia, Amazon Web Services
- Pat Gaw, consultor de segurança sênior, Amazon Web Services
- Daniel Begimher, consultor sênior, segurança, Amazon Web Services
- Danny Cortegaca, arquiteto sênior de soluções de segurança, Amazon Web Services
- Ana Malhotra, arquiteta de soluções de segurança, Amazon Web Services
- Debashis Das, diretor sênior no Escritório do CISO, Amazon Web Services
- Reef Dsouza, arquiteto-chefe de soluções, Amazon Web Services
- Brad Burnett, arquiteto de soluções de segurança, Identity, Amazon Web Services
- Anna McAbee, arquiteta sênior de soluções de segurança, detecção de ameaças e resposta a incidentes, Amazon Web Services
- Jason Garman, arquiteto principal de soluções de segurança, Amazon Web Services

Leitura adicional

Para obter mais ajuda, consulte as seguintes fontes:

- [Whitepaper do AWS Well-Architected Framework](#)
- [Centro de Arquitetura da AWS](#)

Revisões do documento

Para ser notificado sobre atualizações deste whitepaper, inscreva-se no RSS feed.

Alteração	Descrição	Data
Whitepaper atualizado	Práticas recomendadas atualizadas com novas orientações para implementação.	June 27, 2024
Orientação sobre práticas recomendadas atualizada	As práticas recomendadas foram atualizadas com novas orientações nas seguintes áreas: Operar as workloads com segurança e Proteger os dados em trânsito .	December 6, 2023
Orientação sobre práticas recomendadas atualizada	Principais atualizações nas orientações e nas práticas recomendadas na Resposta a incidentes . Várias práticas recomendadas atualizadas na Preparação . Duas novas áreas adicionadas à Resposta a incidente s: Operações e Atividades pós-incidente . Nova prática recomendada adicionada: SEC10-BP08 Estabeleça uma estrutura para aprender com os incidentes .	October 3, 2023
Orientação sobre práticas recomendadas atualizada	As práticas recomendadas foram atualizadas com novas	July 13, 2023

	orientações nas seguintes áreas: Preparar e Simular .	
Atualizações para o novo Framework.	Práticas recomendadas atualizadas com orientações prescritivas e novas práticas recomendadas adicionadas. Nova área de práticas recomendadas de segurança de aplicações (AppSec) adicionada.	April 10, 2023
Whitepaper atualizado	Práticas recomendadas atualizadas com novas orientações para implementação.	December 15, 2022
Whitepaper atualizado	Práticas recomendadas ampliadas e planos de melhoria adicionados.	October 20, 2022
Atualização secundária	Informações do IAM atualizadas para refletir as práticas recomendadas atuais.	June 28, 2022
Atualização secundária	Informações adicionais do AWS PrivateLink incluídas e links quebrados corrigidos.	May 19, 2022
Atualização secundária	AWS PrivateLink adicionado.	May 6, 2022
Atualização secundária	Remoção de linguagem não inclusiva.	April 22, 2022
Atualização secundária	Adicionadas informações sobre o Analisador de Acesso à Rede VPC.	February 2, 2022

Atualização secundária	Pilar Sustentabilidade adicionado à introdução.	December 2, 2021
Atualização secundária	Link quebrado corrigido.	May 27, 2021
Atualização secundária	Alterações editoriais de modo geral.	May 17, 2021
Atualização principal	Seção adicionada sobre governança, detalhes adicionados a várias seções, novos recursos e serviços adicionados.	May 7, 2021
Atualização secundária	Links atualizados.	March 10, 2021
Atualização secundária	Link quebrado corrigido.	July 15, 2020
Atualizações para a nova estrutura de trabalho	Orientações atualizadas sobre gerenciamento de contas, identidades e permissões.	July 8, 2020
Atualizações para a nova estrutura de trabalho	Atualizado para estender orientações a todas as áreas, novas práticas recomendadas, serviços e recursos.	April 30, 2020
Whitepaper atualizado	Atualizações para refletir novos serviços e recursos da AWS e referências atualizadas.	July 1, 2018
Whitepaper atualizado	Seção Configuração e manutenção da segurança do sistema atualizada para refletir os novos serviços e recursos da AWS.	May 1, 2017

Publicação inicial

Publicação do “Pilar
Segurança: AWS Well-Arch
itected Framework”.

November 1, 2016

Avisos

Os clientes são responsáveis por fazer sua própria avaliação independente das informações neste documento. Este documento: (a) é fornecido apenas para fins informativos, (b) representa as práticas e ofertas de produtos atuais da AWS, que estão sujeitas a alterações sem aviso prévio, e (c) não cria nenhum compromisso ou garantia da AWS e suas afiliadas, fornecedores ou licenciadores.

Os produtos ou serviços da AWS são fornecidos “no estado em que se encontram” sem garantias, declarações ou condições de nenhum tipo, explícitas ou implícitas. As responsabilidades e obrigações da AWS para com seus clientes são regidas por contratos da AWS, e este documento não modifica nem faz parte de nenhum contrato entre a AWS e seus clientes.

© 2021, Amazon Web Services, Inc. ou suas afiliadas. Todos os direitos reservados.