Manual do usuário

Agente de segurança da AWS

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Agente de segurança da AWS: Manual do usuário

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens comerciais da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestigie a Amazon. Todas as outras marcas comerciais que não são propriedade da Amazon pertencem aos respectivos proprietários, os quais podem ou não ser afiliados, estar conectados ou ser patrocinados pela Amazon.

Table of Contents

Introdução 1
O que é o AWS Security Agent? 1
Capacidades gerais 1
Benefícios 3
Capacidades do AWS Security Agent 4
Como funciona o AWS Security Agent 5
Visão geral do 5
Entenda a hierarquia de recursos e o ciclo de vida 9
O que é compartilhado em toda a sua organização 9
Qual é o escopo por espaço do agente 10
Como os GitHub repositórios se encaixam na hierarquia 12
Principais diferenças entre os recursos de segurança 12
Entendendo as relações de recursos 14
Conceitos básicos 16
Inscreva-se para AWS 16
Escolha seu caminho. 16
Início rápido: faça um teste de penetração 17
Pré-requisitos 18
Etapa 1: configurar o AWS Security Agent no console da AWS 18
Etapa 2: habilite o teste de penetração e verifique seu domínio 19
Etapa 3: Conectar o código-fonte (opcional) 20
Etapa 4: criar e executar um teste de penetração 20
Etapa 5: revisar os resultados do teste de penetração 21
Início rápido: execute uma revisão de código 22
Pré-requisitos 18
Etapa 1: configurar o AWS Security Agent no console da AWS 18
Etapa 2: ativar e configurar a revisão de código 23
Etapa 3: criar e executar uma revisão de código 25
Etapa 4: Analisar os resultados da revisão de código 25
Início rápido: execute um modelo de ameaça 26
Etapa 1: configurar o AWS Security Agent no console da AWS 18
Etapa 2: Conectar o código-fonte 27
Etapa 3: criar e executar um modelo de ameaça 28
Etapa 4: Analise as ameaças 29

Para administradores (console) 30
Exemplo de configuração 30
Configurar e criar Agent Spaces 30
Configurar o AWS Security Agent 30
Crie um espaço de agente 36
Integrações Connect 38
Como as integrações funcionam com os Agent Spaces 38
Conecte o AWS Security Agent aos GitHub repositórios 40
Remover uma GitHub integração 46
Conecte o AWS Security Agent aos GitLab repositórios 48
Conecte o AWS Security Agent ao GitLab Self-Managed 52
Remover uma GitLab integração 55
Conecte o AWS Security Agent ao servidor GitHub corporativo 56
Remover uma integração com o GitHub Enterprise Server 60
Encontre seu ID de instalação da Atlassian 61
Conecte o AWS Security Agent aos repositórios do Bitbucket 62
Remover uma integração com o Bitbucket 66
Conecte o AWS Security Agent ao Confluence 67
Remover uma integração com o Confluence 71
Conecte-se ao controle de origem hospedado de forma privada 72
Conecte o agente aos recursos privados da VPC 77
Configurar recursos 80
Ativar teste de penetração 80
Habilitar revisão de código de pull request para GitHub repositórios 87
Ativar revisão de código 87
Ative a modelagem de ameaças 94
Permita que os usuários iniciem a remediação dos resultados do teste de penetração e da revisão de código 98
Forneça recursos de agente a partir de um bucket S3 99
Habilite um domínio de aplicativo para testes de penetração 100
Domínios-alvo gerenciados usados para testes de penetração 103
Gerencie os requisitos de segurança 104
Gerencie os requisitos de segurança 104
Pacotes gerenciados pela AWS 111
Gere requisitos de segurança a partir de documentos 113
Preparar documentos para geração de requisitos 115

Gerencie usuários e acessos 116
Conceda aos usuários acesso ao aplicativo web AWS Security Agent 116
Crie uma função do IAM para o AWS Security Agent 118
Chaves gerenciadas pelo cliente 125
Solução de problemas 150
Acesso negado: Tipo de GitHub conta incorreto selecionado ou nome incorreto da organização especificado 150
Acesso negado: permissões insuficientes para instalar o GitHub aplicativo na organização 150
O agente não consegue se conectar ao endpoint durante um teste de penetração 151
Obter ajuda adicional 152
Para usuários e desenvolvedores (aplicativo web e IDE) 153
Faça login no aplicativo web AWS Security Agent 153
Métodos de acesso 153
Faça login com o IAM Identity Center (SSO) 154
Faça login com acesso de administrador (IAM) 155
Crie uma revisão de design 156
Pré-requisitos 18
Etapa 1: comece a criar uma revisão de design 156
Etapa 2: Nomeie sua revisão de design 157
Etapa 3: fazer upload de arquivos de design 157
Etapa 4: iniciar a revisão do projeto 158
Próximas etapas 35
Revise os resultados de uma revisão de design 159
Crie um modelo de ameaça 163
Pré-requisitos 18
Como o AWS Security Agent executa um modelo de ameaça 164
Acesse a página de modelos de ameaças 164
Crie um modelo de ameaça 165
Execute um modelo de ameaça 167
Monitore a execução de um modelo de ameaça 167
Edite um modelo de ameaça 169
Excluir um modelo de ameaça 169
Próximas etapas 35
Analise as ameaças a partir de um modelo de ameaça 170
Analise as descobertas de segurança do código em pull requests 176

Como a revisão de código funciona em pull requests 176
Entendendo os resultados da revisão de código 177
Respondendo às descobertas de segurança 178
Filtrando os resultados da revisão de código 178
Próximas etapas 35
Crie uma revisão de código 180
Pré-requisitos 18
Acesse a página de análises de código 181
Crie uma revisão de código 182
Execute uma revisão de código 187
Monitore uma execução de revisão de código 188
Próximas etapas 35
Analise os resultados de uma revisão de código 189
Corrija as descobertas da revisão de código 196
Execute uma varredura de código diferencial com o S3 199
Como funcionam os escaneamentos diferenciais 199
Pré-requisitos 18
Etapa 1: gerar um arquivo diff 201
Etapa 2: faça o upload do diff para o S3 201
Etapa 3: iniciar um trabalho de revisão de código diferencial 201
Etapa 4: monitorar a digitalização 202
Etapa 5: Analisar as descobertas 203
Cotas e limites 203
Próximas etapas 35
Execute escaneamentos de segurança de código a partir do seu IDE 204
Como funciona a integração com o IDE 204
Pré-requisitos 18
Instale o servidor MCP 206
First-time configuração 207
Execute uma verificação de segurança completa 208
Execute uma varredura diferencial 208
Execute uma análise do modelo de ameaças 209
Revisar descobertas 210
Aplique correções automatizadas 210
Sugestões automáticas de escaneamento (Kiro) 211
Tipos de escaneamento disponíveis 211

Variáveis de ambiente 212
Solução de problemas 212
Cotas e limites 203
Próximas etapas 35
Crie um teste de penetração 214
Pré-requisitos 18
Comece a criar um teste de penetração 214
Nomeie seu teste de penetração 215
Configurar o escopo do teste de penetração 215
Configurar a função do IAM 218
Remediação automática de código 197
Configurar recursos de VPC (opcional) 219
Configurar credenciais de autenticação (opcional) 221
Anexar recursos adicionais (opcional) 223
Crie o teste de penetração 226
Analise os resultados de um teste de penetração 227
Forneça credenciais de autenticação para testes de penetração 239
Corrija a descoberta de um teste de penetração 244
Segurança e referência 247
Segurança 247
Considerações sobre segurança 247
Políticas gerenciadas pela AWS 255
Proteção de dados 258
Gerenciamento de identidade e acesso 262
Resposta a incidentes 278
Validação de conformidade 280
Resiliência 281
Segurança da infraestrutura 282
Endpoints da VPC de interface 283
Análise de configuração e vulnerabilidade 287
Cross-service prevenção delegada confusa 288
Práticas recomendadas de segurança 288
Migração de ações e recursos do IAM 292
Fazendo login com CloudTrail 297
Service Quotas 299
Cotas de operações 299

Cotas de configuração 300
Histórico do documento 300
..... ccciv

Introdução

Saiba o que o AWS Security Agent faz, como funciona e como seus recursos se encaixam.

O que é o AWS Security Agent?

O AWS Security Agent é um agente de fronteira que protege proativamente seus aplicativos em todo o ciclo de vida do desenvolvimento. Ele conduz análises de segurança automatizadas adaptadas às suas necessidades organizacionais, cria modelos de ameaças para identificar como seus aplicativos podem ser atacados e fornece testes de penetração com reconhecimento de contexto sob demanda. Ao validar continuamente a segurança do projeto à implantação, o AWS Security Agent ajuda a evitar vulnerabilidades precocemente em todos os seus ambientes.

As equipes de segurança definem os requisitos de segurança organizacional uma vez no AWS Console: bibliotecas de autorização aprovadas, padrões de registro e políticas de acesso a dados. O AWS Security Agent aplica automaticamente esses requisitos de segurança durante todo o desenvolvimento, avaliando documentos e códigos arquitetônicos de acordo com seus padrões e fornecendo orientação específica ao detectar violações. Isso proporciona uma fiscalização de segurança consistente para análises de design e código em todas as equipes.

Para validação da implantação, o AWS Security Agent transforma o teste de penetração de um gargalo periódico em um recurso sob demanda. As equipes de segurança fornecem URLs de destino, detalhes de autenticação, código-fonte e documentação do aplicativo. O AWS Security Agent desenvolve um profundo entendimento de aplicativos e executa cadeias de ataque sofisticadas para descobrir e validar vulnerabilidades, permitindo que as equipes testem sempre que necessário.

Capacidades gerais

O AWS Security Agent fornece recursos de segurança abrangentes que abrangem todo o ciclo de vida do desenvolvimento.

Análise de segurança do design

O AWS Security Agent transfere a segurança para a esquerda fornecendo feedback de segurança em tempo real sobre documentos de design e avaliando a conformidade com os requisitos de segurança organizacional antes que qualquer código seja escrito. As equipes de segurança

carregam documentos por meio do aplicativo web e recebem orientações de correção para priorizar as descobertas, acelerando as demoradas revisões manuais em análises focadas. Ao incorporar proativamente seus padrões de segurança em cada revisão de projeto, você reduz o retrabalho arquitetônico em estágio avançado e acompanha o ritmo de várias equipes de desenvolvimento.

Modelagem de ameaças

O AWS Security Agent cria modelos de ameaças de seus aplicativos para identificar como eles podem ser atacados. Forneça documentos de design técnico (documentos de escopo) para definir em que o agente se concentra, o código-fonte como contexto para o agente entender seu sistema existente ou ambos. O AWS Security Agent gera uma visão geral do sistema descrevendo a arquitetura, os componentes, os limites de confiança, os fluxos de dados e a postura de segurança do seu aplicativo, junto com um conjunto de ameaças classificadas pela categoria STRIDE (falsificação, adulteração, repúdio, divulgação de informações, negação de serviço, elevação de privilégio) com níveis de gravidade e recomendações acionáveis. Os modelos de ameaças são configurações reutilizáveis que você pode executar novamente à medida que seu código e design evoluem, permitindo a avaliação iterativa da segurança em todo o ciclo de vida do desenvolvimento.

Análise de segurança do código

O AWS Security Agent protege aplicativos de forma proativa por meio de duas abordagens complementares. Primeiro, você pode criar análises de código no aplicativo web para realizar varreduras abrangentes de seu código-fonte completo a partir de repositórios GitHub, GitLab, Bitbucket ou GitHub Enterprise Server ou buckets S3, identificando vulnerabilidades de segurança e validando a conformidade com seus requisitos organizacionais em toda a sua base de código. Em segundo lugar, você pode habilitar a análise automática de pull request para repositórios conectados, onde o AWS Security Agent analisa as alterações de código e publica descobertas de segurança diretamente como comentários de pull request ou merge request. Em ambos os casos, os desenvolvedores recebem orientações de remediação, e o AWS Security Agent pode gerar automaticamente pull requests ou mesclar solicitações com correções de código para vulnerabilidades identificadas. Isso incorpora a experiência em segurança em todos os repositórios, reduzindo os atrasos relacionados à segurança no pipeline de desenvolvimento e escalando a avaliação em todas as bases de código.

Teste de penetração

O AWS Security Agent oferece testes de penetração sob demanda, implantando agentes de IA especializados para descobrir, validar, relatar e corrigir vulnerabilidades de segurança por meio de

cenários de ataque personalizados em várias etapas. O agente entende o contexto do seu aplicativo analisando o código-fonte e a documentação para identificar e explorar vulnerabilidades que as ferramentas automatizadas de verificação de segurança não conseguem encontrar. Ele documenta as descobertas com análise de impacto, caminhos de ataque reproduzíveis e cria pull requests com correções de código prontas para implementar, transformando avaliações periódicas em validação contínua que abrange todos os aplicativos, em vez de se limitar apenas aos críticos.

Benefícios

On-demand acessibilidade

O AWS Security Agent fornece acesso imediato à experiência em segurança. As equipes de desenvolvimento podem realizar análises de design, modelos de ameaças, análises de código e testes de penetração sob demanda sempre que necessário, acompanhando o ritmo dos ciclos de desenvolvimento modernos e permitindo a segurança proativa em todas as etapas.

Valide os requisitos organizacionais automaticamente

Defina os requisitos de segurança da sua organização uma vez no AWS Console. O AWS Security Agent valida automaticamente esses requisitos em todos os aplicativos durante cada revisão de design e segurança do código, garantindo que as equipes atendam aos seus requisitos específicos em vez de listas de verificação genéricas.

Expanda a experiência em segurança

O AWS Security Agent escala as análises de segurança para corresponder à velocidade de desenvolvimento, automatizando a aplicação dos requisitos de segurança organizacional. Configure os requisitos de forma centralizada, conduza análises abrangentes com análises de resultados e gerencie os escopos dos testes de penetração em toda a sua organização por meio do AWS Console.

Correções acionáveis para vulnerabilidades confirmadas

O AWS Security Agent valida as descobertas de segurança encontradas durante o teste de penetração por meio de exploração baseada em provas, fornecendo caminhos de exploração reproduzíveis, análise de impacto abrangente e cria pull requests com correções prontas para implementar em uma linguagem amigável ao desenvolvedor. Isso ajuda as equipes a se concentrarem nos riscos legítimos de segurança de alto impacto sem perder tempo com falsos positivos.

Suporte para nuvem múltipla e híbrida

O AWS Security Agent opera em ambientes AWS, locais, híbridos, multicloud e SaaS, garantindo orientação e testes de segurança consistentes, independentemente de suas escolhas de infraestrutura ou plataforma.

Capacidades do AWS Security Agent

O AWS Security Agent fornece quatro recursos principais de segurança em todo o seu ciclo de vida de desenvolvimento:

- Análise de segurança do projeto — revisa os documentos de design e arquitetura para identificar riscos de segurança. Faça upload de documentos por meio do aplicativo web e o serviço os analisará de acordo com os requisitos de segurança organizacional. O AWS Security Agent avalia a conformidade com seus requisitos de segurança definidos, como bibliotecas de autorização aprovadas, padrões de registro e políticas de acesso a dados. Isso ajuda você a detectar designs inseguros e violações de políticas logo no início do processo de desenvolvimento.
- Modelagem de ameaças — cria um modelo de ameaça do seu aplicativo para identificar como ele pode ser atacado. Forneça documentos de design como documentos de escopo para definir o foco da análise, o código-fonte como contexto para o agente entender seu sistema existente ou ambos. O AWS Security Agent gera uma visão geral do sistema que descreve a arquitetura, os componentes, os limites de confiança, os fluxos de dados e a postura de segurança do seu aplicativo, junto com um conjunto de ameaças classificadas pela categoria STRIDE (falsificação, adulteração, repúdio, divulgação de informações, negação de serviço, elevação de privilégio) com níveis de gravidade e recomendações acionáveis. Cada ameaça está vinculada às evidências em seu código-fonte.
- Análise de segurança de código — analisa o código em seus repositórios e fontes do S3 para identificar vulnerabilidades de segurança e violações dos requisitos de segurança organizacional em várias linguagens, estruturas e arquiteturas. Crie revisões de código no aplicativo web para realizar varreduras abrangentes de seu código-fonte completo ou habilite os comentários do pull request para revisar automaticamente as alterações no GitHub código. O AWS Security Agent fornece orientações específicas de remediação e pode gerar automaticamente pull requests com correções de código para vulnerabilidades identificadas. Isso garante a aplicação consistente de suas políticas de segurança em todas as equipes de desenvolvimento.
- On-demand teste de penetração — descobre, valida, relata e corrige vulnerabilidades de segurança em aplicativos web ativos e APIs por meio de cenários de ataque personalizados em várias etapas. Configure o serviço para criar um pentest por meio do aplicativo web especificando

o escopo do teste, os detalhes da autenticação e os recursos. O AWS Security Agent desenvolve o contexto do aplicativo a partir do código-fonte e da documentação fornecidos e executa cadeias de ataque sofisticadas para identificar vulnerabilidades exploráveis que a análise estática e as ferramentas convencionais ignoram. Ele também fornece correções de código prontas para implementar e cria pull requests diretamente no seu repositório de código, permitindo que você resolva vulnerabilidades ainda mais rápido.

Como funciona o AWS Security Agent

O AWS Security Agent opera em três interfaces para fornecer segurança proativa de aplicativos em todo o ciclo de vida do desenvolvimento. As configurações de segurança são gerenciadas no AWS Management Console, as análises de segurança são conduzidas no aplicativo Web Security Agent e a remediação automatizada de códigos e descobertas de segurança ocorre diretamente em plataformas de repositório de código, como. GitHub

Visão geral do

O AWS Security Agent consiste em três componentes principais:

- AWS Management Console — configure Agent Spaces, defina requisitos de segurança, configure testes de penetração, conecte repositórios de código e gerencie o acesso de usuários
- Aplicativo Web Security Agent - Conduza análises de design, crie e execute modelos de ameaças, execute testes de penetração, crie e execute análises de código e analise as descobertas de segurança nos Agent Spaces atribuídos
- Integração do repositório de código - Receba análises automatizadas de código em pull requests e pull requests de remediação de testes de penetração. Atualmente, o AWS Security Agent oferece suporte à conexão GitHub a.

Você trabalha com o AWS Security Agent em uma das três funções, que você pode identificar pela interface que você usa:

Perfil	Onde você trabalha e o que você faz
Administrador	Funciona no AWS Management Console — configura Agent Spaces, define requisitos de segurança, configura recursos e gerencia o acesso do usuário.

Perfil	Onde você trabalha e o que você faz
Usuário	Funciona no aplicativo Web do Security Agent — executa análises de design, modelos de ameaças, testes de penetração e análises de código, além de analisar as descobertas resultantes.
Desenvolvedor	Funciona em GitHub um IDE (como Kiro ou Claude Code) — recebe descobertas de segurança automatizadas em pull requests e executa varreduras de código sem sair do ambiente de desenvolvimento.

Usamos esses nomes de funções por toda parte para indicar quem executa cada tarefa. Uma única pessoa pode ter mais de uma função.

Configuração do console

Os administradores configuram o AWS Security Agent por meio do AWS Management Console.

Agent Spaces — Quando você cria seu primeiro espaço de agente no AWS Management Console, o AWS Security Agent cria o aplicativo web para sua conta. Cada Agent Space que você cria representa um aplicativo ou projeto distinto que você deseja proteger. No aplicativo Web, os usuários selecionam em qual espaço do agente trabalhar ao realizar avaliações de segurança.

Requisitos de segurança — defina os requisitos de segurança que o AWS Security Agent avalia durante as análises de design e código, organizados em pacotes de requisitos de segurança. Habilite AWS-managed pacotes com base nos padrões do setor, crie pacotes personalizados para as políticas da sua organização ou gere requisitos fazendo o upload de sua documentação de segurança. Os requisitos se aplicam a todos os Agent Spaces.

Configuração do teste de penetração - Configure os recursos do teste de penetração para cada espaço do agente ao:

- Verificando domínios de destino para testes por meio de verificação de DNS ou HTTP
- Configurando o acesso à VPC para testar aplicativos privados
- Configurando o CloudWatch registro para execuções de testes de penetração
- Configuração das funções do AWS Secrets Manager ou Lambda para credenciais de teste
- Especificação de buckets do S3 para contexto adicional do aplicativo

Configuração de revisão de código - Configure os recursos de revisão de código para cada espaço do agente da seguinte forma:

- Conectando GitHub repositórios ou buckets do S3 contendo código-fonte
- Selecionar configurações de revisão de código (vulnerabilidades de segurança, requisitos personalizados ou ambos)
- Habilitando comentários de pull request para análise automática de alterações de código no GitHub
- Configurando o CloudWatch registro para execuções de revisão de código

Configuração de modelagem de ameaças - Configure os recursos de modelagem de ameaças para cada espaço de agente ao:

- Conectando repositórios de código-fonte ou buckets S3 contendo código-fonte
- Adicionar documentos de escopo (documentos de design de recursos) para focar a análise em um recurso específico
- Configurando o CloudWatch registro para execuções de modelos de ameaças
- Configurando uma função de serviço para acesso a recursos da AWS

Integrações - Conecte GitHub repositórios a cada espaço do agente para habilitar os principais recursos:

- Forneça o contexto do aplicativo para testes de penetração mais precisos
- Habilite a revisão de código para escaneamentos completos do repositório e análise de pull request
- Habilite a remediação automatizada de código por meio de pull requests para análise de código e descobertas de testes de penetração

Acesso do usuário - Gerencie como os usuários acessam o aplicativo Web do Security Agent. Se você habilitou o IAM Identity Center (SSO), atribua usuários no console do AWS Security Agent para fornecer acesso direto por SSO ao aplicativo web. Se você estiver usando o IAM-only acesso, os usuários com acesso ao AWS Console podem iniciar o aplicativo web por meio do link de acesso de administrador no console para qualquer espaço de agente.

Atividades de aplicativos da Web

Os usuários acessam o aplicativo Web do Security Agent para realizar avaliações de segurança nos Agent Spaces atribuídos.

Selecione Espaço do Agente - Ao fazer login no Aplicativo Web, os usuários selecionam em qual Espaço do Agente trabalhar. Os usuários só podem ver e acessar os Agent Spaces aos quais foram atribuídos.

Análises de design - Faça upload de documentos de design e especificações de arquitetura para análise em relação aos requisitos de segurança organizacional. Analise as descobertas com orientações de remediação.

Modelos de ameaças - Crie e execute modelos de ameaças fornecendo código-fonte, documentos de design técnico (documentos de escopo) ou ambos. Os documentos de escopo definem em que o agente concentra sua análise; o código-fonte fornece contexto sobre seu sistema existente. Cada execução produz uma visão geral do sistema descrevendo a arquitetura, os limites de confiança, os fluxos de dados e a postura de segurança do seu aplicativo, junto com um conjunto de ameaças classificadas por categoria STRIDE com níveis de gravidade e recomendações acionáveis.

Revisões de código - Crie e execute análises de código que realizam análises estáticas abrangentes em todo o código-fonte. Selecione GitHub repositórios ou fontes do S3, defina as configurações de escaneamento e analise as descobertas detalhadas de segurança com orientações de remediação. O AWS Security Agent identifica vulnerabilidades de segurança e valida a conformidade com os requisitos de segurança personalizados da sua organização em toda a sua base de código.

Testes de penetração - configure e execute testes de penetração fornecendo URLs de destino, detalhes de autenticação e documentação. O AWS Security Agent realiza testes autônomos para descobrir vulnerabilidades exploráveis por meio de cenários de ataque em várias etapas.

Analise as descobertas - Examine as descobertas de segurança detalhadas de análises de design, modelos de ameaças, análises de código e testes de penetração, incluindo análise de impacto, caminhos de ataque reproduzíveis e orientação de remediação.

Valide as correções — avaliações Re-run de segurança após a implementação de correções para verificar se as vulnerabilidades foram resolvidas.

GitHub integração

O AWS Security Agent se integra diretamente GitHub para fornecer feedback de segurança automatizado nos fluxos de trabalho dos desenvolvedores.

Comentários de pull request — Depois que os administradores instalam o GitHub aplicativo AWS Security Agent e habilitam a revisão de código com comentários de revisão de código para um espaço de agente, o AWS Security Agent analisa automaticamente as pull requests em repositórios conectados. Os desenvolvedores recebem descobertas de segurança e orientações sobre remediação diretamente nos comentários do pull request, validando as alterações no código em relação aos requisitos de segurança organizacional e às vulnerabilidades comuns.

Remediação automática — Quando os usuários habilitam a remediação automática de código para uma revisão de código, o AWS Security Agent gera correções para vulnerabilidades identificadas e envia pull requests aos repositórios associados. GitHub

Remediação do teste de penetração - Quando os administradores ativam a busca de remediação no console, os usuários podem solicitar a remediação automática das descobertas do teste de penetração no aplicativo Web. O AWS Security Agent abre uma pull request para o GitHub repositório associado com correções de código para resolver a vulnerabilidade.

Entenda a hierarquia de recursos e o ciclo de vida

O AWS Security Agent organiza recursos de testes de segurança em uma estrutura hierárquica que determina o que é compartilhado em toda a sua organização e qual é o escopo por aplicativo. A compreensão dessa estrutura ajuda você a configurar o AWS Security Agent de forma eficaz e a saber onde encontrar e gerenciar diferentes recursos.

O que é compartilhado em toda a sua organização

Alguns recursos no AWS Security Agent são configurados uma vez no nível organizacional e se aplicam a todos os seus aplicativos e Agent Spaces. Esses recursos em nível de locatário fornecem consistência e reduzem o trabalho duplicado de configuração.

Recurso	O que é	Por que é compartilhado
Requisitos de segurança	Padrões de segurança organizacional que definem o que o AWS Security Agent valida durante as análises de design e código	Suas políticas de segurança se aplicam a todos os aplicativos. Defina-os uma vez e o AWS Security Agent os aplicará em todos os lugares.

Recurso	O que é	Por que é compartilhado
GitHub integrações	GitHub Organizações registradas ou contas de usuário autorizadas a se conectar ao AWS Security Agent	Registre sua GitHub organização uma vez e, em seguida, conecte repositórios específicos a qualquer Espaço do Agente, conforme necessário.
Configurações do IAM Identity Center	Configurações de SSO que controlam como os usuários acessam o AWS Security Agent	O gerenciamento centralizado de identidades se aplica a todos os Agent Spaces da sua organização.

> **Important**
>
> As mudanças nos requisitos de segurança afetam todas as futuras revisões de design e revisões de código em todos os Agent Spaces. As avaliações existentes não são afetadas.

Qual é o escopo por espaço do agente

Cada espaço do agente representa um aplicativo ou projeto distinto que você deseja proteger. Os recursos no nível do Agent Space são direcionados para esse aplicativo específico, permitindo que diferentes equipes trabalhem de forma independente com suas próprias configurações e avaliações.

Recurso	O que é	Por que seu escopo é definido por aplicativo
Configurações de teste de penetração	Teste configurações para recursos específicos, endpoints de API ou funcionalidades em seu aplicativo	Cada aplicativo tem alvos exclusivos, métodos de autenticação e limites de escopo específicos para esse aplicativo.
Revisões de design	Avaliações individuais de segurança arquitetônica de documentos de projeto	Cada aplicativo tem sua própria arquitetura e documentos de design que são avaliados de forma independente.

Recurso	O que é	Por que seu escopo é definido por aplicativo
Modelos de ameaças	Avaliações de modelagem de ameaças que criam uma visão geral do sistema e identificam ameaças a partir do código-fonte, dos documentos de design ou de ambos	Cada aplicativo tem seu próprio código e design e é modelado de forma independente contra ameaças. Os modelos de ameaças são configurações reutilizáveis que você pode executar novamente à medida que seu código e design evoluem.
Integrações	Provedores de origem e documentação (GitHub GitLab,, Bitbucket , GitHub Enterprise Server e Confluence) conectados a este Agent Space	Aplicativos diferentes dependem de fontes e documentações diferentes. Conectá-los no nível do Agent Space mantém os limites do aplicativo claros.
Configurações de revisão de código	Configuração de recursos de revisão de código, incluindo fontes conectadas, configurações de digitalização e ativação de comentários de relações públicas	Cada aplicativo tem seus próprios repositórios e as necessidades de análise de segurança são configuradas de forma independente.
Configurações de remediação do teste de penetração	Configuração de quais repositórios conectados podem receber solicitações automáticas de correção para resultados de testes de penetração	As equipes controlam onde o AWS Security Agent pode enviar alterações de código com base no fluxo de trabalho do aplicativo.
Atribuições de usuários	Usuários que têm acesso a esse espaço de agente específico	As equipes só veem avaliações de segurança dos aplicativos pelos quais são responsáveis, mantendo o trabalho organizado e focado.

> **Tip**
>
> Recomendamos criar um espaço de agente por aplicativo ou projeto para manter limites claros entre as equipes e organizar as avaliações de segurança de forma eficaz.

Como os GitHub repositórios se encaixam na hierarquia

GitHub os repositórios são integrados por meio de um processo de várias etapas que conecta recursos organizacionais a aplicativos específicos:

1. Registre-se no nível de locatário - Autorize o GitHub aplicativo AWS Security Agent para sua GitHub organização ou conta de usuário uma vez
2. Conecte-se no nível do Espaço do Agente - Selecione repositórios específicos para se conectar a cada Espaço do Agente
3. Configure o uso por repositório - habilite recursos específicos para cada repositório conectado:
   - Revisão de código - Escaneamento completo do código-fonte e análise automatizada de pull request
   - Contexto do teste de penetração - Compreensão do aplicativo a partir do código-fonte durante os testes de penetração
   - Remediação automática de código - pull requests automatizados com correções de vulnerabilidades para descobertas de análise de código e testes de penetração

Um único repositório pode ser conectado a vários Agent Spaces com diferentes recursos habilitados em cada um.

Principais diferenças entre os recursos de segurança

Cada recurso de segurança no AWS Security Agent segue um modelo de fluxo de trabalho diferente com base em como as equipes de segurança o usam.

Teste de penetração: configurações reutilizáveis com execuções independentes

Os testes de penetração usam um modelo de configuração e execução que oferece suporte a testes de segurança iterativos:

- Crie uma vez, execute várias vezes - Defina uma configuração para um destino específico (endpoint de API, área de recursos) com limites de escopo, autenticação e parâmetros de teste
- Execuções independentes - Execute a mesma configuração várias vezes à medida que melhora a segurança. Cada execução é independente e gera novas descobertas

Esse modelo oferece suporte à validação contínua da segurança à medida que você desenvolve e implementa melhorias.

Revisões de design: One-off avaliações com clonagem

As análises de design são avaliações independentes que não seguem um modelo de configuração reutilizável:

- Avaliação única - Cada revisão de design analisa os documentos enviados uma vez em relação aos requisitos de segurança da sua organização
- Não é possível executar novamente - as revisões de design não são reutilizáveis. Você não pode executar novamente a mesma avaliação
- Clonar para atualizações - clone uma revisão de design existente para criar uma nova revisão com os documentos originais pré-carregados, permitindo que você atualize documentos e execute uma nova análise

Esse modelo oferece suporte a avaliações pontuais de segurança arquitetônica.

Revisões de código: configurações reutilizáveis com varreduras sob demanda e análise automática de relações públicas

As análises de código fornecem dois modos de operação para proteger seu código-fonte:

- Revisões completas de código (aplicativo web) - Crie configurações de revisão de código que selecionem GitHub repositórios ou fontes do S3 e, em seguida, execute varreduras abrangentes sob demanda. Cada execução executa análises estáticas em todo o código-fonte e gera descobertas com orientações de remediação. Você pode executar novamente a mesma configuração de revisão de código à medida que seu código evolui.
- Comentários do pull request (GitHub) - Habilite a análise automatizada para GitHub repositórios conectados. O AWS Security Agent analisa automaticamente as pull requests quando elas são marcadas como prontas para análise e publica as descobertas de segurança como comentários diretamente no site GitHub.

Ambos os modos usam suas configurações de revisão de código definidas (vulnerabilidades de segurança, requisitos personalizados ou ambos) e oferecem suporte à correção automática de código por meio de pull requests.

Modelos de ameaças: configurações reutilizáveis com execuções sob demanda

Os modelos de ameaças usam um modelo de configuração e execução que oferece suporte à avaliação iterativa de sua arquitetura:

- Crie uma vez, execute várias vezes — defina um modelo de ameaça selecionando o código-fonte como fonte, carregando documentos de design como documentos de escopo ou ambos. Execute-o sob demanda e execute-o novamente à medida que seu código e design evoluem.
- Entradas flexíveis — Execute um modelo de ameaça somente no código-fonte, somente nos documentos de design ou em ambos. Os documentos de escopo definem em que o agente concentra sua análise; o código-fonte fornece contexto sobre seu sistema existente.
- Visão geral do sistema e ameaças — Cada execução produz uma visão geral do sistema descrevendo a arquitetura, os limites de confiança, os fluxos de dados e a postura de segurança do seu aplicativo, junto com um conjunto de ameaças classificadas pela categoria STRIDE com severidade, evidências e recomendações práticas.

Entendendo as relações de recursos

A hierarquia determina onde você configura e acessa recursos diferentes:

No AWS Management Console:

- Configurar recursos em nível de inquilino (requisitos de segurança, GitHub integrações, IAM Identity Center)
- Crie e gerencie Agent Spaces
- Definir as configurações do Agent Space (repositórios conectados, habilitação de revisão de código, remediação de testes de penetração)

No aplicativo Web do Security Agent:

- Crie e gerencie configurações de testes de penetração e execuções de testes
- Crie e gerencie revisões de design
- Crie, gerencie e execute análises de código em repositórios conectados e fontes do S3

- Crie, gerencie e execute modelos de ameaças com base no código-fonte, nos documentos de escopo ou em ambos
- Veja os resultados de testes de penetração, análises de código, análises de design e modelos de ameaças

Em GitHub:

- Veja os resultados da revisão do código do pull request como comentários do pull request
- Receba pull requests automatizados de remediação para resultados de análise de código e testes de penetração (quando ativados no Agent Space)

> **Note**
>
> Os resultados da revisão do código do pull request aparecem em GitHub. Os resultados completos da análise do código, do teste de penetração e da revisão do design aparecem no aplicativo Web do Security Agent.

Conceitos básicos

Inscreva-se, encontre o ponto de partida certo para sua meta e faça sua primeira avaliação com um guia de início rápido.

Inscreva-se para um AWS account

Para começar AWS, você precisa de uma AWS conta. Para obter informações sobre como criar uma AWS conta, consulte Introdução a uma AWSAWS conta no Guia de referência de gerenciamento de contas.

Escolha seu caminho.

Encontre o ponto de partida certo para o que você quer fazer com o AWS Security Agent. Use a tabela abaixo para combinar sua meta com uma tarefa e, em seguida, vá diretamente para essa página. Se você é novo no AWS Security Agent, leia the section called “O que é o AWS Security Agent?” primeiro para obter uma visão geral rápida.

Eu quero...	Quem	Comece aqui
Entenda o que o AWS Security Agent faz antes de configurá-lo	Todos	the section called “O que é o AWS Security Agent?”
Configure o serviço e crie um espaço para agentes	Administrador	the section called “Configurar o AWS Security Agent”
Teste meu aplicativo ativo ou implantado em busca de vulnerabi lidades exploráveis	Usuário	the section called “Início rápido: faça um teste de penetração”
Escaneie meu código-fonte em busca de vulnerabilidades e violações de políticas	Usuário	the section called “Início rápido: execute uma revisão de código”
Modele como meu aplicativo pode ser atacado (STRIDE)	Usuário	the section called “Início rápido: execute um modelo de ameaça”

Eu quero...	Quem	Comece aqui
Receba feedback de segurança sobre um design antes de eu escrever o código	Usuário	the section called “Crie uma revisão de design”
Digitalize o código sem sair do meu IDE (Kiro ou Claude Code)	Desenvolvedor	the section called “Execute escaneamentos de segurança de código a partir do seu IDE”
Digitalize somente minhas linhas alteradas antes de mesclar	Desenvolvedor	the section called “Execute uma varredura de código diferencial com o S3”
Receba comentários de segurança automáticos sobre pull requests e solicitações de mesclagem	Administrador	the section called “Ativar revisão de código”
Analise as descobertas e decida o que corrigir primeiro	Usuário	Teste de penetração, revisão de código, modelo de ameaça, revisão de design

> **Note**
>
> O AWS Security Agent usa três funções, que você pode identificar pela interface na qual você trabalha: o administrador trabalha no AWS Management Console (instalação e configuração), o usuário trabalha no aplicativo web (executando avaliações e revisando as descobertas) e o desenvolvedor trabalha em um IDE, como Kiro GitHub ou Claude Code. Uma única pessoa pode ter mais de uma função. Para obter as definições completas, consultethe section called “Como funciona o AWS Security Agent”.

Início rápido: faça um teste de penetração

Este guia de início rápido mostra como executar seu primeiro teste de penetração (pentest) com o AWS Security Agent. Um teste de penetração exercita seu aplicativo implantado em relação a um domínio de destino verificado e retorna descobertas de segurança, cada uma com uma classificação

de severidade e evidências de apoio. Ele abrange um aplicativo acessível ao público; para testar um hospedado em uma VPC privada, consulte. the section called “Ativar teste de penetração”

> **Note**
>
> Você precisa acessar o AWS Management Console para configurar o AWS Security Agent e definir o escopo do teste, além de acessar o aplicativo web para criar e executar testes de penetração.

Pré-requisitos

Antes de começar, você deve ter o seguinte:

- Acesso ao AWS Management Console com permissões para configurar o AWS Security Agent.
- Um domínio de destino ativo que hospeda o aplicativo que você deseja testar.
- A capacidade de adicionar um registro DNS para esse domínio ou uma zona hospedada do Route 53 na mesma conta da AWS, para que você possa verificar a propriedade.
- (Opcional) Um repositório de código-fonte (GitHub GitLab, ou Bitbucket) para seu aplicativo, para dar mais contexto ao agente.

Etapa 1: configurar o AWS Security Agent no console da AWS

Se você ainda não configurou o AWS Security Agent, conclua a configuração inicial:

1. Navegue até o AWS Security Agent no AWS Management Console.
2. Selecione Configurar o AWS Security Agent.
3. Crie um espaço de agente. Um espaço de agente pode ser usado por vários usuários e deve ser específico para cada aplicativo que você deseja testar. Insira um nome e uma descrição. O nome deve identificar o aplicativo que você deseja testar de penetração.
4. Selecione IAM-only acesso em Configuração de acesso do usuário.
   - Esse guia de início rápido não abrange a habilitação do single sign-on (SSO) com o IAM Identity Center, que permite que os usuários acessem o aplicativo web diretamente do AWS Console.
   - Para permitir que usuários sem acesso ao AWS Management Console iniciem testes de penetração, habilite a integração do IAM Identity Center. Para obter detalhes, consulte the section called “Conceda aos usuários acesso ao aplicativo web AWS Security Agent”.

5. Escolha Configurar o AWS Security Agent.

> **Note**
>
> Quando você escolhe Configurar, o AWS Security Agent cria seu espaço de agente e estabelece um aplicativo web onde os usuários podem executar testes de penetração, análises de código, modelos de ameaças e análises de design.

Etapa 2: habilite o teste de penetração e verifique seu domínio

> **Note**
>
> No console da AWS, você define o escopo do que pode ser testado. Em seguida, os usuários executam testes de penetração específicos dentro desse escopo a partir do aplicativo da web.

1. Na barra lateral esquerda, selecione Agent Spaces e, em seguida, selecione o Agent Space que você criou na Etapa 1.
2. Selecione a guia Teste de penetração e escolha Configurar teste de penetração para abrir o assistente.
3. Configurar domínio - Insira o domínio de destino que você deseja testar e selecione um método de verificação, registro DNS TXT ou rota HTTP. O domínio deve estar ativo e hospedar o aplicativo que você deseja testar. Escolha Próximo.
4. Verificar domínios - Verifique a propriedade de cada domínio na tabela Domínios de destino. O AWS Security Agent executa testes somente em domínios verificados. Para obter etapas detalhadas e os valores exatos a serem copiados, consultethe section called “Habilite um domínio de aplicativo para testes de penetração”.
   - Domínios do Route 53 na mesma conta da AWS - Selecione o domínio e escolha a One-click verificação. O AWS Security Agent cria o registro DNS e conclui a verificação para você.
   - Registro DNS TXT (outros provedores de DNS) - Na tabela Domínios de destino, copie o nome do registro e o valor do registro, adicione-os como um registro TXT com seu provedor de DNS, selecione o domínio e escolha Verificar. As alterações de DNS podem levar algum tempo para se propagar, portanto, a verificação pode não ser concluída imediatamente.

- Rota HTTP - Crie um arquivo `.well-known/aws/securityagent-domain-verification.json` em seu servidor web, adicione o token de verificação no formato`{"tokens": ["<token>"]}`, selecione o domínio e escolha Verificar.

5. (Opcional) Configurar recursos adicionais - Adicione recursos opcionais, como grupos de CloudWatch registros e credenciais. A seção de acesso ao serviço é pré-configurada: o AWS Security Agent cria uma função de serviço com as permissões necessárias, a menos que você escolha uma função do IAM existente.

Etapa 3: Conectar o código-fonte (opcional)

Conectar um provedor de código-fonte fornece ao AWS Security Agent um contexto sobre seu aplicativo e melhora a cobertura dos testes de penetração. Esta etapa é opcional.

1. Na guia Teste de penetração, escolha Adicionar no banner Conectar um provedor de código-fonte para teste de penetração na parte superior da página.
2. Registre e conecte seu provedor e, em seguida, selecione os repositórios a serem disponibilizados para testes de penetração.

Para ver o fluxo completo de integração, consulte Conectar o AWS Security Agent aos GitHub repositórios ou o tópico de conexão para seu provedor.

Etapa 4: criar e executar um teste de penetração

> **Note**
>
> Você cria e executa testes de penetração no aplicativo web do AWS Security Agent. Recomendamos a execução de testes em um ambiente de teste que espelhe de perto a produção.

1. Inicie o aplicativo Web: selecione a guia Aplicativo Web e, em seguida, Acesso de administrador.
2. Na barra lateral esquerda, escolha Testes de penetração e, em seguida, Criar um teste de penetração.
3. Detalhes do teste de penetração - Configure o escopo do teste e a fonte de registro:
   a. Insira um nome de Pentest.

b. Em URLs de destino, selecione ou insira um domínio verificado para testar (por exemplo,`https://example.com`). Somente domínios verificados podem ser testados, e os subdomínios são cobertos automaticamente.

c. (Opcional) Em URLs acessíveis, adicione domínios que o agente deve acessar para fazer login e navegar, mas não deve atacar, como provedores de identidade, CDNs ou APIs fora do seu domínio de destino. O AWS Security Agent pode alcançar esses domínios, mas não os testa; somente seus domínios de destino são atacados.

d. Analise as regras de configuração de rede para confirmar quais endpoints podem ou não receber tráfego durante o teste.

e. Em Permissões, selecione uma função de serviço e, opcionalmente, um grupo de CloudWatch registros. Escolha Próximo.

4. (Opcional) Recursos de VPC — configure uma VPC se seu destino estiver em uma rede privada que não seja acessível ao público. Consulte the section called “Ativar teste de penetração”.
5. (Opcional) Recursos de autenticação - Forneça credenciais para que o agente possa acessar caminhos autenticados e não públicos. Recomendamos adicioná-los para ampliar a cobertura e revelar vulnerabilidades por trás de um login.
6. (Opcional) Configuração adicional - adicione contexto do aplicativo, como arquivos, GitHub repositórios ou fontes do S3, para melhorar a qualidade da busca. Você também pode ativar a correção automática de código e definir outras opções de execução aqui.
7. Escolha Criar e executar para iniciar o teste agora ou Criar pentest para salvá-lo e executá-lo posteriormente.

Etapa 5: revisar os resultados do teste de penetração

1. Um teste de penetração pode levar várias horas para ser concluído. A maioria é concluída em 16 horas, dependendo do tamanho e da complexidade da sua aplicação.
2. A execução começa com uma fase de Preflight que valida a configuração antes do início do teste: ela configura a infraestrutura de registro, verifica se os endpoints de destino estão acessíveis e prepara o ambiente de teste. Se uma verificação de pré-voo falhar (por exemplo, um domínio de destino que não pode ser resolvido), a execução é interrompida antes do início do teste. Abra a guia Preflight para ver qual verificação falhou, resolva-a e inicie uma nova execução.
3. Quando a execução for concluída, abra-a e revise as guias Visão geral da execução, Registros e Descobertas.

4. Na guia Descobertas, selecione uma descoberta para ver sua descrição, gravidade, tipo de risco e evidências de apoio.

Para obter mais detalhes, consulte the section called “Crie um teste de penetração” e the section called “Analise os resultados de um teste de penetração”.

Início rápido: execute uma revisão de código

Este guia de início rápido orienta você na execução de sua primeira revisão de código com o AWS Security Agent. O AWS Security Agent verifica seus repositórios de código-fonte em busca de vulnerabilidades de segurança e conformidade com os requisitos de segurança da sua organização.

> **Note**
>
> Você precisa acessar o AWS Management Console para configurar o AWS Security Agent e acessar o aplicativo web para criar e executar análises de código.

Pré-requisitos

Antes de começar, você deve ter o seguinte:

- Acesso ao AWS Management Console com permissões para configurar o AWS Security Agent.
- Um repositório de código-fonte (GitHub GitLab, ou Bitbucket) ou uma fonte do Amazon S3 que contém o código que você deseja revisar.

Etapa 1: configurar o AWS Security Agent no console da AWS

Se você ainda não configurou o AWS Security Agent, conclua a configuração inicial:

1. Navegue até o AWS Security Agent no AWS Management Console.
2. Selecione Configurar o AWS Security Agent.
3. Crie um espaço de agente. Um espaço de agente pode ser usado por vários usuários e deve ser específico para cada aplicativo que você deseja testar. Insira um nome e uma descrição para seu primeiro espaço de agente. Esse nome aparece para os usuários no aplicativo web. O nome deve identificar o aplicativo cujo código você deseja revisar.
4. Selecione IAM-only acesso em Configuração de acesso do usuário.

- Este guia de início rápido não abrange a habilitação do single sign-on (SSO) com o IAM Identity Center. Isso permite que os usuários acessem diretamente o aplicativo web do AWS Security Agent, a partir do AWS Console.
- Para permitir que usuários sem acesso ao AWS Management Console executem análises de código, habilite a integração do IAM Identity Center. Para obter detalhes, consulte the section called “Conceda aos usuários acesso ao aplicativo web AWS Security Agent”.

5. Escolha Configurar o AWS Security Agent.

> **Note**
>
> Quando você escolhe Configurar, o AWS Security Agent cria seu espaço de agente e estabelece um aplicativo web onde os usuários podem executar testes de penetração, análises de código, modelos de ameaças e análises de design.

Etapa 2: ativar e configurar a revisão de código

> **Note**
>
> Se você já tem GitHub repositórios ou buckets S3 conectados ao seu Espaço do Agente (por exemplo, por meio da configuração do teste de penetração), a revisão de código já está habilitada. Você pode pular essa etapa e ir diretamente para o aplicativo da web.

Abra o assistente de configuração da revisão de código

1. Na barra lateral esquerda, selecione Agent Spaces e, em seguida, selecione seu Agent Space.
2. Selecione Habilitar revisão de código no cabeçalho ou na guia Revisão de código.

Connect integrações, repositórios e buckets

1. (Se você ainda não tem uma GitHub integração) Crie um GitHub registro. Se você já tiver um, vá para a próxima etapa.
   a. Na seção Integrações conectadas, escolha Adicionar e, em seguida, Criar novo registro.
   b. Selecione GitHube escolha Avançar.

c. Escolha Instalar e autorizar e conclua a instalação em: GitHub

i. Selecione o GitHub usuário ou a organização que possui o repositório que você deseja revisar.

ii. Selecione Todos os repositórios ou Selecione Somente repositórios.

iii. Escolha Instalar e Autorizar e conclua a GitHub autenticação.

d. De volta ao AWS Management Console, insira um nome de registro e confirme se o tipo de conta corresponde ao local em que você instalou o GitHub aplicativo.

e. Escolha Connect para salvar o registro.

Para ver o fluxo completo de GitHub integração, consulte Connect AWS Security Agent aos GitHub repositórios.

2. Conecte GitHub repositórios. Na seção Integrações conectadas, escolha Adicionar e selecione seu GitHub registro. O GitHub assistente Connect em duas etapas é aberto:

a. Em Connect GitHub repositories, selecione os repositórios a serem incluídos e escolha Avançar.

b. Em Gerenciar recursos, alterne o seguinte por repositório:

- Comentários de revisão de código — Permita que o AWS Security Agent publique descobertas de segurança como comentários sobre pull requests no repositório.
- Remediação automática — Permita que os usuários do aplicativo web do AWS Security Agent solicitem pull requests que corrijam as descobertas.

c. Escolha Salvar para retornar ao assistente de configuração.

3. (Opcional) Conecte as fontes do S3. Na seção S3 buckets, escolha Adicionar recurso S3 e insira o URI do S3 para um bucket contendo código-fonte, ou escolha Browse para escolher um.

4. Selecione suas configurações de revisão de código. O padrão, Requisitos de segurança e descobertas de vulnerabilidades, analisa o código quanto à conformidade com os requisitos de segurança que você habilitou e às vulnerabilidades comuns.

5. Escolha Próximo.

Configurações opcionais

1. Configure grupos de CloudWatch registros opcionais e acesso ao serviço. A função de serviço padrão é pré-configurada com as permissões necessárias.

2. Escolha Salvar.

Etapa 3: criar e executar uma revisão de código

> **Note**
>
> Você cria e executa análises de código somente no aplicativo web do AWS Security Agent.

1. Selecione a guia Aplicativo Web e, em seguida, Acesso de administrador para iniciar o aplicativo web do AWS Security Agent.
2. Na barra lateral esquerda, escolha Revisões de código.
3. Escolha Criar revisão de código.
4. Configure a revisão do código:
   a. Insira um título que identifique o escopo dessa análise (por exemplo, “revisão de segurança do serviço de cobrança”).
   b. Em Fontes, selecione os GitHub repositórios que você conectou ao seu Espaço do Agente na Etapa 2 ou insira as fontes do S3 que você deseja verificar.
   c. Selecione a função de serviço nas funções configuradas.
   d. (Opcional) Selecione Habilitar a remediação automática de código para que o AWS Security Agent envie automaticamente pull requests com correções para todas as descobertas.
5. Escolha Criar revisão de código.
6. Na página de detalhes da revisão de código, escolha Iniciar revisão.

Etapa 4: Analisar os resultados da revisão de código

1. A revisão do código normalmente leva de 30 a 60 minutos, dependendo do tamanho da sua base de código.
2. A execução começa com uma fase de Preflight que valida a configuração antes do início da verificação: ela confirma que o AWS Security Agent pode extrair seu código-fonte do seu repositório ou fonte do S3 e configurar o ambiente de digitalização. Se uma verificação de pré-voo falhar (por exemplo, ela não conseguir acessar sua fonte), a execução será interrompida antes da análise estática. Abra a guia Preflight para ver qual verificação falhou, resolva-a e inicie uma nova execução.
3. Depois de concluído, navegue até a execução concluída e selecione a guia Descobertas.
4. Analise as descobertas na visualização detalhada da lista:

a. Selecione uma descoberta no painel esquerdo para ver seus detalhes.

b. Revise as seções Descrição, Localizações do código, Evidência e Correção sugerida.

c. Use o código de correção para gerar uma pull request com uma correção ou revise os PRs de remediação automática se você habilitou essa opção.

Para obter mais detalhes, consulte the section called “Crie uma revisão de código” e the section called “Analise os resultados de uma revisão de código”.

Início rápido: execute um modelo de ameaça

Este guia de início rápido mostra como executar seu primeiro modelo de ameaça com o AWS Security Agent. Um modelo de ameaça analisa a arquitetura do seu aplicativo e produz uma visão geral do sistema (como o AWS Security Agent entende seu sistema) e um conjunto de ameaças (como ele pode ser atacado, cada uma com um nível de gravidade, classificação STRIDE e recomendações). Você pode executar um modelo de ameaça em documentos de design (documentos de escopo) para definir o foco, código-fonte (fontes) para fornecer contexto sobre seu sistema existente ou ambos.

> **Note**
>
> Você precisa acessar o AWS Management Console para configurar o AWS Security Agent e acessar o aplicativo web para criar e executar modelos de ameaças.

Etapa 1: configurar o AWS Security Agent no console da AWS

Se você ainda não configurou o AWS Security Agent, conclua a configuração inicial:

1. Navegue até o AWS Security Agent no AWS Management Console.
2. Selecione Configurar o AWS Security Agent.
3. Crie um espaço de agente. Um espaço de agente pode ser usado por vários usuários e deve ser específico para cada aplicativo que você deseja proteger. Insira um nome e uma descrição para seu primeiro espaço de agente. O nome deve identificar o aplicativo que você deseja ameaçar como modelo.
4. Selecione IAM-only acesso em Configuração de acesso do usuário.

- Este guia de início rápido não abrange a habilitação do single sign-on (SSO) com o IAM Identity Center. Isso permite que os usuários acessem diretamente o aplicativo web do AWS Security Agent a partir do AWS Console.
- Se você quiser permitir que usuários sem acesso ao AWS Management Console realizem tarefas como iniciar um modelo de ameaça, você deve habilitar a integração do IAM Identity Center.

5. Escolha Configurar o AWS Security Agent.

> **Note**
>
> Quando você escolhe Configurar, o AWS Security Agent cria seu espaço de agente e estabelece um aplicativo web onde os usuários podem executar testes de penetração, análises de código, modelos de ameaças e análises de design.

Etapa 2: Conectar o código-fonte

> **Note**
>
> Se você já tem repositórios ou buckets S3 conectados ao seu Espaço do Agente (por exemplo, por meio de revisão de código ou configuração de teste de penetração), você pode pular essa etapa e ir diretamente para o aplicativo web. Você sempre pode executar um modelo de ameaça em documentos de escopo enviados sem conectar nenhum código-fonte.

Conecte repositórios ou buckets do S3 que contêm o código-fonte que você deseja que o agente use como contexto para entender seu sistema existente:

1. Na barra lateral esquerda, selecione Agent Spaces e, em seguida, selecione seu Agent Space.
2. Navegue até a seção Integrações para conectar seu provedor de código-fonte.
3. Como alternativa, conecte buckets do S3 que contêm seu código-fonte.

Para ver o fluxo completo de integração, consulte Connect AWS Security Agent aos GitHub repositórios.

Etapa 3: criar e executar um modelo de ameaça

> **Note**
>
> Você cria e executa modelos de ameaças no aplicativo web do AWS Security Agent.

1. Inicie o aplicativo web na guia Aplicativo web no AWS Management Console. Como alternativa, se você tiver o IAM Identity Center configurado, faça login diretamente.
2. Na barra lateral esquerda, escolha Modelos de ameaças.
3. Escolha Criar modelo de ameaça.
4. Configure o modelo de ameaça:
   a. Insira um título que identifique o escopo desse modelo de ameaça (por exemplo, “billing-service” ou “checkout-api”).
   b. Forneça pelo menos uma entrada:
      - Em Documentos de escopo, faça upload de documentos de design (DOC, DOCX, JPEG, MD, PDF, PNG ou TXT), insira URIs do S3 ou selecione páginas do Confluence.
      - Em Fontes, selecione repositórios de suas integrações conectadas ou insira URIs do S3 contendo o código-fonte.

      > **Tip**
      >
      > Forneça documentos de origem e escopo para definir o modelo de ameaça em um design específico (por exemplo, um documento de design de recursos) e, ao mesmo tempo, forneça ao agente seu código-fonte como contexto para entender seu sistema existente.

   c. Selecione a função de serviço nas funções configuradas.
   d. (Opcional) Selecione um grupo de CloudWatch registros para registros.
5. Escolha Criar modelo de ameaça.
6. Na página de detalhes do modelo de ameaça, escolha Iniciar execução.

Etapa 4: Analise as ameaças

1. A execução avança em suas tarefas de análise. Você pode monitorar o progresso na guia Preflight e visualizar os detalhes da tarefa na guia Registros.
2. Depois de concluído, o status da execução muda para Concluído.
3. Selecione a guia Visão geral para revisar a visão geral do sistema e a distribuição da gravidade.
4. Selecione a guia Ameaças para analisar as ameaças identificadas:
   a. Selecione uma ameaça na lista para ver seus detalhes no painel lateral.
   b. Analise a Declaração, a Gravidade, as categorias STRIDE, a Recomendação, a Evidência e outros campos.
   c. Atualize o status da ameaça para Resolvida ou Descartada à medida que você aborda ou faz a triagem de cada ameaça.

Para obter mais detalhes, consulte the section called “Crie um modelo de ameaça” e the section called “Analise as ameaças a partir de um modelo de ameaça”.

Para administradores (console)

Como administrador, você configura o AWS Security Agent no AWS Management Console e configura os Agent Spaces que os usuários acessam por meio do aplicativo web do AWS Security Agent. Cada espaço do agente representa um ambiente distinto com permissões e recursos específicos.

Recomendamos criar um Agent Space exclusivo para cada aplicativo que você deseja testar. Por exemplo, se você tiver dois projetos internos — um aplicativo de cobrança e um aplicativo de controle de tarefas — você deve criar dois Agent Spaces separados.

Exemplo de configuração

Considere a possibilidade de um administrador configurar um Espaço do Agente para avaliar a segurança de um aplicativo de cobrança interno. O administrador poderia:

- Verifique o domínio (como`beta.billing.example.com`)
- Conecte-se GitHub e ative a revisão de código
- Configure o acesso à rede atribuindo uma VPC, sub-rede e grupo de segurança apropriados para testes de penetração

Quando os usuários iniciam um teste de penetração ou uma revisão do projeto, eles podem selecionar esses recursos pré-configurados, trabalhando dentro das barreiras que você definiu e, ao mesmo tempo, mantendo a flexibilidade para suas necessidades específicas de avaliação.

Configurar e criar Agent Spaces

Configure o AWS Security Agent para sua organização e crie os Agent Spaces que definem o escopo dos recursos e avaliações de cada aplicativo.

Configurar o AWS Security Agent

Configure o AWS Security Agent para sua organização criando seu primeiro espaço de agente e estabelecendo como os usuários acessarão o aplicativo web.

Essa configuração inicial permite que os administradores configurem recursos de segurança no console da AWS e fornece aos usuários acesso à análise do projeto e ao teste de penetração por

meio do aplicativo web Security Agent. Após essa configuração única, você pode criar espaços de agente adicionais com um processo simplificado.

Visão geral do

Entendendo os espaços de agentes

Um Espaço do Agente é um espaço de trabalho dedicado para proteger um aplicativo ou projeto específico. Ele contém todas as análises de segurança, GitHub repositórios opcionalmente conectados, configurações de testes de penetração, resultados e descobertas desse aplicativo.

Os Agent Spaces ajudam você a organizar o trabalho de segurança mantendo as avaliações de segurança de cada aplicativo separadas, permitindo que as equipes se concentrem em seus aplicativos específicos. Recomendamos criar um espaço de agente por aplicativo ou projeto para manter limites claros.

Os requisitos de segurança são definidos no nível da organização e se aplicam a todos os Agent Spaces, enquanto cada Agent Space mantém seus próprios documentos de design, repositórios de código, configurações de testes de penetração, resultados de testes de penetração e descobertas de segurança.

O que está incluído em um Espaço do Agente

Cada Espaço do Agente contém:

- GitHub Repositórios conectados opcionalmente associados ao aplicativo ou projeto
- Análises anteriores de design do aplicativo
- Configurações e limites de testes de penetração específicos para a aplicação
- Resultados do teste de penetração e descobertas de segurança

O que você configurará durante a configuração

Durante essa configuração inicial, você tomará decisões organizacionais que se aplicam a todos os Agent Spaces:

- Método de acesso — Escolha como os usuários acessarão o aplicativo web do Security Agent (IAM Identity Center SSO) ou IAM-only acessarão por meio do AWS Console)
- Permissões - Configure a função do IAM que o aplicativo web usa para acessar os serviços da AWS

- Primeiro Espaço do Agente - Crie seu Espaço do Agente inicial com um nome e uma descrição

> **Note**
>
> Depois de concluir essa configuração, criar Agent Spaces adicionais é mais simples. Basta fornecer um nome e uma descrição. Consulte the section called “Crie um espaço de agente” para obter detalhes sobre a criação de Agent Spaces subsequentes.

Pré-requisitos

Antes de começar, verifique se você tem:

- Permissões para criar e gerenciar recursos do AWS Security Agent
- Compreensão dos requisitos de gerenciamento de identidade da sua organização
- (Opcional) Conta da AWS com permissões para criar funções do IAM e configurar o IAM Identity Center (se estiver usando o acesso SSO)
- (Opcional) Função existente do IAM se você quiser usar uma configuração de permissões personalizada

Etapa 1: Crie seu primeiro Espaço do Agente

Defina as propriedades básicas do seu primeiro Agent Space que serão exibidas aos usuários no aplicativo web.

1. Na página do console do AWS Security Agent, escolha Configurar o Security Agent.
2. No campo Nome do Espaço do Agente, insira um nome para o seu Espaço do Agente.

> **Note**
>
> O nome do Espaço do Agente é exibido aos usuários no aplicativo web e ajuda a identificar qual aplicativo ou projeto esse espaço representa.

3. (Opcional) No campo Descrição, forneça uma descrição que ajude a distinguir a finalidade do Espaço do Agente.

> **Tip**
>
> A descrição ajuda a distinguir o propósito do Agent Space. Recomendamos descrever o aplicativo ou projeto específico que esse Agent Space protegerá, como “Aplicativo web do portal do cliente”, “Microsserviços de processamento de pagamentos” ou “Plataforma de análise interna”.

Etapa 2: escolha seu método de acesso

Selecione como os usuários acessarão o aplicativo Web do Security Agent. Você escolhe entre habilitar o acesso por SSO por meio do IAM Identity Center ou fornecer acesso por meio do AWS Console.

> **Note**
>
> Se você escolher o IAM-only acesso e depois quiser usar o IAM Identity Center, precisará excluir a configuração do AWS Security Agent e concluir o processo de configuração novamente.

1. Na seção Método de acesso, selecione uma das seguintes opções:
   - IAM Identity Center (SSO) — Habilite o acesso por SSO para sua equipe por meio do IAM Identity Center. Essa opção é recomendada para equipes que precisam de gerenciamento centralizado de usuários e desejam que os usuários acessem o aplicativo web diretamente sem passar pelo AWS Console.
   - IAM-only access — Forneça acesso por meio de um link de acesso de administrador no AWS Console. Essa opção é mais simples de configurar e adequada para equipes que preferem acesso baseado em console ou não precisam de recursos de SSO.
2. Se você selecionou o IAM Identity Center (SSO), continue na Etapa 2a abaixo.
3. Se você selecionou IAM-only acesso, vá para a Etapa 3.

Etapa 2a: Configurar o IAM Identity Center (somente acesso por SSO)

Conclua essas etapas somente se você tiver selecionado o IAM Identity Center (SSO) como seu método de acesso.

1. Na seção Connect to IAM Identity Center, analise a região da AWS.

> **Important**
>
> Seu aplicativo deve estar configurado na mesma região em que você habilita o IAM Identity Center. A região exibida é onde seu Espaço do Agente será criado. O IAM Identity Center deve estar ativado na mesma região em que você cria seu espaço de agente.

2. Na seção IAM Identity Center, escolha uma das seguintes opções:
   - Escolha Criar instância de conta para criar uma nova instância de conta do IAM Identity Center
   - Se uma instância da organização já existir na mesma região, o AWS Security Agent se conectará automaticamente a ela

> **Note**
>
> Depois que o AWS Security Agent estiver conectado ao IAM Identity Center, você pode gerenciar o acesso atribuindo usuários no IAM Identity Center ao aplicativo.

3. Se você estiver conectando o Active Directory ou um provedor de identidade externo, revise o alerta informativo.

> **Important**
>
> Se você já está gerenciando usuários no Active Directory ou em outra fonte de identidade e planeja conectar essa fonte de identidade ao IAM Identity Center, cancele a configuração e acesse primeiro o console do IAM Identity Center. Escolha a fonte de identidade que você deseja conectar antes de configurar o AWS Security Agent. Alterar as fontes de identidade posteriormente pode remover as atribuições de usuários existentes.

Etapa 3: configurar permissões (opcional)

Configure a função do IAM que seu aplicativo web do Security Agent usa para acessar serviços, APIs e contas da AWS.

1. Localize a seção Configuração de permissões - opcional.
2. Se a seção estiver fechada, escolha a seção Configuração de permissões para expandi-la.
3. Selecione uma das opções a seguir:

- Criar função padrão — O AWS Security Agent cria automaticamente uma nova função do IAM com as permissões necessárias para o aplicativo web
- Use outra função - selecione uma função do IAM existente entre as opções disponíveis

4. Analise a descrição da função:

> **Note**
>
> Uma função padrão do IAM será criada para que seu aplicativo web acesse outros serviços, APIs e contas da AWS. A função receberá as permissões necessárias para operações de avaliação de segurança.

Etapa 4: Concluir a configuração

Depois de definir todas as configurações necessárias, conclua a configuração para criar seu primeiro Espaço do Agente e estabelecer o Aplicativo Web do Security Agent.

1. Revise todas as seções de configuração para garantir a precisão.
2. Escolha Configurar.
3. O AWS Security Agent cria seu espaço de agente, estabelece a aplicação web e configura os recursos necessários da AWS.

> **Note**
>
> Após a configuração inicial, você terá a opção de configurar recursos, incluindo limites de testes de penetração, requisitos de segurança e GitHub integração.

Próximas etapas

Depois de configurar o AWS Security Agent:

- Defina os requisitos de segurança para sua organização
- Configure recursos de teste de penetração, incluindo verificação de domínio e acesso VPC para seu Agent Space
- Conecte GitHub repositórios para análise de código e contexto de teste de penetração

- (Se estiver usando o IAM Identity Center) Atribua usuários ao espaço do agente no console do AWS Security Agent
- (Se estiver usando o IAM-only acesso) Inicie o aplicativo web por meio do link de acesso do administrador no console da AWS
- Crie espaços de agente adicionais para outros aplicativos (consultethe section called “Crie um espaço de agente”)

Crie um espaço de agente

Crie Agent Spaces para proteger aplicativos ou projetos em sua organização. Cada Espaço do Agente fornece um espaço de trabalho para avaliações, descobertas e configurações de segurança específicas desse aplicativo ou projeto e pode ser acessado por vários usuários.

Esse procedimento pressupõe que você já tenha concluído a configuração inicial do AWS Security Agent. Se você ainda não configurou o AWS Security Agent, consultethe section called “Configurar o AWS Security Agent”.

Visão geral do

Entendendo os espaços de agentes

Um Espaço do Agente é um espaço de trabalho dedicado para proteger um aplicativo ou projeto específico. Ele contém todas as análises de segurança, repositórios de código opcionalmente conectados, configurações de testes de penetração, resultados e descobertas desse aplicativo.

Os Agent Spaces ajudam você a organizar o trabalho de segurança mantendo as avaliações de segurança de cada aplicativo separadas, permitindo que as equipes se concentrem em seus aplicativos específicos. Recomendamos criar um espaço de agente por aplicativo ou projeto para manter limites claros.

Para obter uma explicação abrangente sobre os Agent Spaces e como eles se encaixam na estrutura de segurança da sua organização, consultethe section called “Entenda a hierarquia de recursos e o ciclo de vida”.

O que está incluído em um Espaço do Agente

Cada Espaço do Agente contém:

- Repositórios de código conectados opcionalmente associados ao aplicativo ou projeto

- Análises de design anteriores para o aplicativo ou projeto
- Configurações e limites de testes de penetração específicos para a aplicação
- Resultados do teste de penetração e descobertas de segurança

Crie um espaço de agente

Crie um novo Espaço do Agente para um aplicativo ou projeto que você deseja proteger.

1. No console do AWS Security Agent, navegue até a página Agent Spaces.
2. Escolha Criar espaço do agente.
3. No campo Nome do Espaço do Agente, insira um nome para o seu Espaço do Agente.

> **Note**
>
> O nome do Espaço do Agente é exibido aos usuários no aplicativo web e ajuda a identificar qual aplicativo ou projeto esse espaço representa.

4. (Opcional) No campo Descrição, forneça uma descrição que ajude a distinguir a finalidade do Espaço do Agente.

> **Tip**
>
> A descrição ajuda a distinguir o propósito do Agent Space. Recomendamos descrever o aplicativo ou projeto específico que esse Agent Space protegerá, como “Aplicativo web do portal do cliente”, “Microsserviços de processamento de pagamentos” ou “Plataforma de análise interna”.

5. Escolha Criar.

> **Note**
>
> O AWS Security Agent criará seu espaço de agente. Agora você pode configurar recursos e conectar recursos específicos para esse aplicativo.

Próximas etapas

Depois de criar seu Espaço do Agente:

- Conecte repositórios de código-fonte (GitHub, GitLab, Bitbucket ou GitHub Enterprise Server) para análise de código e contexto de teste de penetração
- Connect Confluence para contextualizar a documentação em análises de design e testes de penetração
- Ative o recurso de revisão de código para repositórios conectados (consultethe section called "Habilitar revisão de código de pull request para GitHub repositórios")
- Configure recursos de teste de penetração, incluindo verificação de domínio
- (Se estiver usando o IAM Identity Center) Atribua usuários a esse Espaço do Agente na seção Aplicativo Web da página Espaço do Agente. (verthe section called "Conceda aos usuários acesso ao aplicativo web AWS Security Agent")
- (Se estiver usando o IAM-only acesso) Os usuários com acesso ao console podem iniciar o aplicativo web por meio do link de acesso de administrador para este Espaço do Agente

Integrações Connect

Conecte e gerencie provedores de código-fonte (GitHub GitLab,, Bitbucket e GitHub Enterprise Server) e provedores de documentação (Confluence), juntamente com a conectividade de rede privada, que seus Agent Spaces usam para análise de código, testes de penetração e modelagem de ameaças.

Como as integrações funcionam com os Agent Spaces

O AWS Security Agent se conecta a fornecedores terceirizados de código-fonte e documentação para que o agente possa analisar seu código e sua documentação durante as avaliações de segurança. Antes de conectar um provedor específico, é útil entender como as integrações se relacionam com os Agent Spaces e os recursos.

Registre-se uma vez, reutilize em qualquer lugar

Conectar um provedor ao AWS Security Agent envolve duas etapas distintas que acontecem em níveis diferentes:

1. Registre a integração (nível da conta). Você registra um provedor uma vez na página de integrações no console de gerenciamento do AWS Security Agent. Um registro representa a conexão autorizada com uma conta de provedor, como uma GitHub organização, um GitLab grupo, um espaço de trabalho do Bitbucket ou um site do Confluence. Os registros são recursos em nível de conta.

2. Conecte recursos a um Espaço do Agente (nível do Espaço do Agente). De dentro de um Agent Space, você conecta recursos de uma integração registrada — repositórios para provedores de código-fonte ou páginas para o Confluence — e configura como o agente os usa. Essa etapa é específica para cada Espaço do Agente.

Um único registro é reutilizado em cada Espaço do Agente em sua conta. Se você registrou um provedor ao configurar um Espaço do Agente, esse mesmo registro estará disponível quando você conectar recursos a outro Espaço do Agente — você não registrará o mesmo provedor novamente.

Uma conexão, compartilhada entre os recursos

Quando você conecta um recurso a um Espaço do Agente, esse recurso é compartilhado entre os recursos do agente. Você não conecta um repositório separadamente para cada recurso.

- O acesso de leitura é concedido conectando o recurso. A conexão de um repositório permite que o AWS Security Agent o leia para varreduras completas de código (revisão de código), contexto de teste de penetração, modelagem de ameaças e revisão de design. Conectar uma página do Confluence permite que o agente a leia para contextualizar a documentação em análises de design, modelagem de ameaças e testes de penetração.
- As ações de gravação são opcionais, por repositório. Ao conectar repositórios de código-fonte a um Espaço do Agente, você também pode habilitar ações de gravação para cada repositório:
  - Comentários de revisão de código — O AWS Security Agent publica os resultados da revisão como comentários sobre pull requests (ou solicitações de mesclagem GitLab).
  - Remediação de código — O AWS Security Agent abre solicitações de pull (ou solicitações de mesclagem) com correções para vulnerabilidades descobertas.

    Essas ações de gravação não estão disponíveis para repositórios públicos. Read-only a análise ainda se aplica.

> **Note**
>
> O Confluence é um provedor de documentação em vez de um provedor de código-fonte. O AWS Security Agent lê páginas conectadas do Confluence para fornecer contexto para análises de design, modelagem de ameaças e testes de penetração. Selecionar uma página concede acesso de leitura; não há alternâncias de capacidade por página.

Provedores compatíveis

O AWS Security Agent oferece suporte aos seguintes provedores:

- Código-fonte - GitHub (Enterprise hospedado na nuvem GitHub e hospedado na nuvem), GitHub GitHub Enterprise Server (auto-hospedado), (hospedado na nuvem), GitLab (auto-hospedado) e Bitbucket GitLab Self-Managed Cloud.
- Documentação - Confluence Cloud.

Para provedores auto-hospedados que não podem ser acessados pela Internet pública, você pode rotear o tráfego do agente por meio de uma conexão privada. Para obter mais informações, consulte the section called “Conecte-se ao controle de origem hospedado de forma privada”.

Próximas etapas

- Registre um provedor na página de integrações. Veja o tópico de conexão do seu provedor, comothe section called “Conecte o AWS Security Agent aos GitHub repositórios”.
- Conecte recursos a um Espaço do Agente e configure os recursos. Consulte the section called “Ativar revisão de código” e the section called “Ativar teste de penetração”.

Conecte o AWS Security Agent aos GitHub repositórios

Conecte seu AWS Security Agent aos GitHub repositórios para permitir recursos de análise de código, modelagem de ameaças, testes de penetração e remediação automatizada. O AWS Security Agent oferece suporte tanto para empresas hospedadas na nuvem GitHub quanto na nuvem GitHub . Antes de começar, analise the section called “Como as integrações funcionam com os Agent Spaces” para entender como um registro é reutilizado nos Agent Spaces e compartilhado entre os recursos.

GitHub a integração serve a vários propósitos:

> **Note**
>
> Esta página abrange empresas hospedadas na nuvem GitHub (github.com) e hospedadas na nuvem. GitHub Para o GitHub Enterprise Server auto-hospedado, consultethe section called “Conecte o AWS Security Agent ao servidor GitHub corporativo”.

- Revisão de código - Analise automaticamente as alterações de código em cada pull request de acordo com seus requisitos de segurança organizacional e execute varreduras de repositório completo sob demanda
- Modelagem de ameaças - Forneça compreensão do aplicativo analisando o código-fonte, os fluxos de dados e a arquitetura
- Contexto do teste de penetração - Forneça compreensão do aplicativo para testes de penetração analisando o código-fonte
- Remediação automatizada - envie pull requests com correções para vulnerabilidades descobertas durante avaliações de segurança

GitHub A conexão com o AWS Security Agent exige a autorização do GitHub aplicativo AWS Security Agent para sua GitHub organização ou conta de usuário e, em seguida, o registro da conexão no Console AWS.

Como GitHub a integração funciona

A análise do pull request acontece dentro de GitHub. Depois de autorizar o GitHub aplicativo, conectar repositórios e habilitar comentários de revisão de código no Console de Gerenciamento da AWS, o AWS Security Agent verifica automaticamente as alterações em cada nova pull request (uma verificação diferencial apenas do código alterado) e publica as descobertas como comentários da pull request.

Você cria e executa análises completas de código — que examinam toda a base de código de um repositório — no aplicativo web do AWS Security Agent, não no. GitHub

Os testes de penetração e a modelagem de ameaças são iniciados no aplicativo web do AWS Security Agent. Os usuários selecionam repositórios conectados para fornecer o contexto do aplicativo e especificar domínios de destino para testes de penetração. Se você habilitar a remediação automatizada, os usuários poderão solicitar que o AWS Security Agent corrija as descobertas abrindo pull requests em repositórios conectados.

Pré-requisitos

Antes de começar, verifique se você tem:

- GitHub acesso de administrador da organização ou acesso do proprietário da conta de GitHub usuário

- Permissões para configurar integrações para seu espaço de agente no AWS Management Console
- Compreender quais repositórios você deseja conectar para análise de código, modelagem de ameaças e testes de penetração

Important

Um GitHub aplicativo só pode ser instalado uma vez em uma GitHub conta ou GitHub organização. Se você precisar conectar a mesma GitHub organização ao AWS Security Agent, deverá usar a mesma conta da AWS em que a integração foi registrada pela primeira vez.

Note

Se sua organização GitHub corporativa habilitou a lista de IPs permitidos, você deverá aceitar os endereços IP permitidos no GitHub aplicativo. Você também pode optar por adicionar automaticamente os endereços IP à sua lista de permissões. Para obter mais informações, consulte Permitir o acesso por GitHub aplicativos e Habilitar endereços IP permitidos na GitHub documentação.
Os seguintes endereços IP são usados para acessar seus GitHub recursos:

- Leste dos EUA (Norte da Virgínia) (us-east-1)
  - `34.228.181.128`
  - `44.219.176.187`
  - `54.226.244.221`
- Oeste dos EUA (Oregon) (us-west-2)
  - `34.212.16.133`
  - `52.89.67.212`
  - `54.187.135.61`
- Ásia-Pacífico (Mumbai) (ap-south-1)
  - `13.126.209.199`
  - `13.234.6.24`
  - `35.154.102.216`

- Ásia-Pacífico (Singapura) (ap-southeast-1)
  - `18.139.13.125`
  - `47.130.240.215`
  - `54.179.238.173`
- Ásia-Pacífico (Sydney) (ap-southeast-2)
  - `13.237.95.197`
  - `13.238.84.102`
  - `52.64.174.242`
- Ásia Pacific (Tóquio) (ap-northeast-1)
  - `13.192.12.233`
  - `35.74.181.230`
  - `57.183.50.158`
- Europa (Frankfurt) (eu-central-1)
  - `18.158.110.140`
  - `52.57.96.160`
  - `52.59.55.56`
- Europa (Irlanda) (eu-west-1)
  - `34.251.85.24`
  - `52.30.157.157`
  - `52.51.192.222`
- América do Sul (São Paulo) (sa-east-1)
  - `54.94.247.213`
  - `54.207.222.14`
  - `54.232.201.242`

Autorize e registre o aplicativo AWS Security Agent GitHub

Autorize o GitHub aplicativo AWS Security Agent a acessar sua GitHub organização ou conta de usuário e, em seguida, registre a conexão no console da AWS.

> **Important**
>
> Conclua todas as etapas desse processo sem fechar o navegador ou sair da navegação. Se o processo de registro for interrompido, talvez seja necessário desinstalar o GitHub aplicativo e começar de novo.

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Escolha Adicionar integração.
3. Selecione GitHub.
4. Escolha Próximo.
5. Escolha Instalar e autorizar.

   Você será redirecionado GitHub para concluir a autorização. Certifique-se de estar conectado GitHub com uma conta que tenha acesso de administrador à organização ou conta de usuário que você deseja conectar.
6. Em GitHub, selecione a conta ou organização na qual você deseja instalar o GitHub aplicativo AWS Security Agent.
7. Selecione quais repositórios o AWS Security Agent pode acessar:
   - Todos os repositórios - Conceda acesso a todos os repositórios atuais e futuros na organização ou na conta do usuário
   - Selecione somente repositórios - Escolha repositórios específicos no menu suspenso. Você pode selecionar vários repositórios, um por vez.

   > **Note**
   >
   > Você pode modificar o acesso ao repositório a qualquer momento acessando as configurações do GitHub aplicativo na sua GitHub organização ou nas configurações da conta do usuário.
8. Escolha Instalar e autorizar.
9. Você será redirecionado de volta ao AWS Management Console para concluir o registro.
10. Na seção Detalhes do registro, configure os seguintes campos:

a. Nome do registro - insira um nome descritivo para essa GitHub conexão. Use um nome que identifique a GitHub organização ou a conta do usuário, como "Acme-Corp-Org" ou "Production-Repos”.

b. Tipo de conta - Selecione uma das seguintes opções no menu suspenso:

- Organização - Se você conectou uma conta GitHub da organização
- Usuário - Se você conectou uma conta de GitHub usuário pessoal

c. Nome da organização (aparece somente se você selecionou Organização) - Insira o nome exato da sua GitHub organização conforme aparece em GitHub.

11. Selecione Conectar.
12. Você vê uma mensagem de confirmação e retorna à página Integrações, onde sua nova GitHub conexão aparece com o nome de registro. Para conectar outras GitHub organizações ou contas de usuário, repita esse processo escolhendo Adicionar integração novamente.

Solucionar problemas de integração GitHub

Se você encontrar problemas durante o processo de GitHub integração, use as diretrizes a seguir para resolver problemas comuns.

Não foi possível concluir o registro

Se você não conseguiu concluir o processo de registro (por exemplo, seu navegador foi fechado, você saiu da página de registro ou teve uma interrupção na sessão), o GitHub aplicativo AWS Security Agent pode estar instalado em sua GitHub organização, mas não registrado no Console da AWS.

Sintomas:

- Quando você tenta autorizar o GitHub aplicativo novamente, GitHub mostra “Configurar” em vez de “Instalar”
- Você não pode concluir o registro no AWS Console
- A integração não aparece na sua lista de integrações

Resolução:

1. Desinstale o GitHub aplicativo AWS Security Agent da sua GitHub organização ou conta de usuário.

2. Retorne ao console do AWS Security Agent e inicie o processo de integração novamente desde o início.

Várias contas da AWS tentando integrar a mesma GitHub organização

Um GitHub aplicativo só pode ser instalado uma vez em uma GitHub conta ou GitHub organização. Se você precisar usar repositórios de uma GitHub organização que já esteja integrada a uma conta diferente do AWS Security Agent, você deve usar a conta da AWS em que a integração foi registrada pela primeira vez.

Resolução:

- Identifique qual conta do AWS Security Agent tem a GitHub integração registrada
- Use essa conta da AWS para criar Agent Spaces e conectar repositórios
- Se você precisar mover a integração para uma conta diferente da AWS, primeiro desinstale o GitHub aplicativo da conta original da AWS e, em seguida, integre-o à nova conta

Próximas etapas

Depois de se conectar GitHub ao AWS Security Agent:

- Navegue até o Espaço do Agente onde você deseja usar esses repositórios
- Escolha Habilitar revisão de código ou Configurar teste de penetração para conectar repositórios específicos ao seu Agent Space e configurar seu uso
- Habilite a remediação automatizada para permitir que o AWS Security Agent envie pull requests com correções de vulnerabilidade
- Revise as permissões do GitHub aplicativo e o acesso ao repositório nas configurações GitHub da sua organização

Remover uma GitHub integração

Remova uma GitHub integração quando você não precisar mais do AWS Security Agent para acessar repositórios de uma GitHub organização ou conta de usuário específica. Você deve primeiro desinstalar o GitHub aplicativo AWS Security Agent GitHub antes de remover a integração no Console AWS.

Pré-requisitos para remoção

Antes de remover uma GitHub integração, verifique se você tem:

- Verificou quais Agent Spaces têm repositórios conectados a partir dessa integração
- Compreendi o impacto: a remoção dessa integração interromperá as conexões do repositório para análise de código, contexto de teste de penetração e remediação de testes de penetração em todos os Agent Spaces usando repositórios dessa integração
- GitHub acesso de administrador da organização ou acesso do proprietário da conta de usuário para desinstalar o GitHub aplicativo

Etapa 1: Desinstalar o GitHub aplicativo AWS Security Agent do GitHub

Primeiro, desinstale o GitHub aplicativo AWS Security Agent da sua GitHub organização ou conta de usuário.

Para GitHub Organizations:

1. Acesse github.com e abra a página da sua organização.
2. Na barra lateral esquerda, escolha Configurações.
3. Em Código, planejamento e automação, escolha GitHub Aplicativos instalados.
4. Localize o aplicativo AWS Security Agent na lista.
5. Escolha Configurar ao lado do aplicativo AWS Security Agent.
6. Role até a parte inferior da página de configuração e escolha Desinstalar.
7. Confirme a desinstalação quando solicitado.

Para contas GitHub de usuário:

1. Acesse github.com.
2. Escolha sua foto de perfil.
3. Escolha Configurações.
4. Na barra lateral esquerda, selecione Aplicativos.
5. Abra a guia GitHub Aplicativos instalados.
6. Localize o aplicativo AWS Security Agent na lista.

7. Escolha Configurar ao lado do aplicativo AWS Security Agent.
8. Role até a parte inferior da página de configuração e escolha Desinstalar.
9. Confirme a desinstalação quando solicitado.

Etapa 2: remover a integração do AWS Security Agent

Depois de desinstalar o GitHub aplicativo, remova o registro de integração do AWS Console.

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Localize a GitHub integração que você deseja remover na lista de integrações.
3. Selecione a integração clicando nela.
4. Escolha Remover .
5. Revise a caixa de diálogo de confirmação, que avisa sobre o impacto:

> ⚠ Warning
>
> A remoção dessa integração afetará todos os Agent Spaces que têm repositórios conectados a partir dessa GitHub organização ou conta de usuário. Isso quebrará:
>
> - Funcionalidade de revisão de código para repositórios conectados
> - Contexto de teste de penetração de repositórios conectados
> - Recursos de remediação de testes de penetração para repositórios conectados
>
> Certifique-se de ter desinstalado o GitHub aplicativo AWS Security Agent GitHub antes de continuar.

6. Se você ainda não desinstalou o GitHub aplicativo GitHub, receberá um aviso. Retorne à Etapa 1 para concluir a desinstalação primeiro.
7. Se você desinstalou o GitHub aplicativo e entende o impacto, escolha Confirmar remoção.
8. A integração é removida da sua lista de integrações. Qualquer Agent Spaces com repositórios dessa integração não tem mais acesso a esses repositórios para análise de código, contexto de teste de penetração ou remediação automatizada.

Conecte o AWS Security Agent aos GitLab repositórios

Conecte seu AWS Security Agent aos repositórios GitLab na nuvem para habilitar recursos de análise de código, modelagem de ameaças, testes de penetração e remediação automatizada. Antes

de começar, analise the section called “Como as integrações funcionam com os Agent Spaces” para entender como um registro é reutilizado nos Agent Spaces e compartilhado entre os recursos.

GitLab a integração serve a vários propósitos:

- Revisão de código - analise automaticamente as alterações de código em cada solicitação de mesclagem de acordo com seus requisitos de segurança organizacional e execute varreduras de repositório completo sob demanda
- Modelagem de ameaças - Forneça compreensão do aplicativo analisando o código-fonte, os fluxos de dados e a arquitetura
- Contexto do teste de penetração - Forneça compreensão do aplicativo para testes de penetração analisando o código-fonte
- Remediação automatizada - envie solicitações de mesclagem com correções para vulnerabilidades descobertas durante avaliações de segurança

GitLab Para se conectar ao AWS Security Agent, é necessário fornecer um token de GitLab acesso com as permissões apropriadas e, em seguida, registrar a conexão no Console da AWS.

Como GitLab a integração funciona

A análise da solicitação de mesclagem ocorre em GitLab. Depois de fornecer seu token de acesso, conectar projetos e habilitar comentários de revisão de código no Console de Gerenciamento da AWS, o AWS Security Agent verifica automaticamente as alterações em cada nova solicitação de mesclagem (uma verificação diferencial apenas do código alterado) e publica as descobertas como comentários da solicitação de mesclagem.

Você cria e executa análises completas de código — que examinam toda a base de código de um projeto — no aplicativo web do AWS Security Agent, não no. GitLab

Os testes de penetração e a modelagem de ameaças são iniciados no aplicativo web do AWS Security Agent. Os usuários especificam domínios de destino e selecionam repositórios conectados para fornecer o contexto do aplicativo. Se você habilitar a remediação automatizada, os usuários poderão solicitar que o AWS Security Agent corrija as descobertas abrindo solicitações de mesclagem em repositórios conectados.

> **Note**
>
> A correção automatizada não está disponível para GitLab repositórios públicos para evitar a divulgação de vulnerabilidades antes que elas sejam corrigidas.

Pré-requisitos

Antes de começar, verifique se você tem:

- Uma GitLab.com conta com acesso de mantenedor ou proprietário aos projetos que você deseja conectar
- Um token de GitLab acesso com os escopos necessários para seu tipo de conexão:
  - Pessoal - Um token de acesso pessoal com todas as permissões de leitura e a `api` permissão.
  - Grupo - Um token de acesso de grupo com os `read_api` `read_repository` escopos e.
- Permissões para configurar integrações no console de gerenciamento do AWS Security Agent

> **Important**
>
> Defina a expiração do token para um máximo de 365 dias após a data atual.

> **Note**
>
> GitLab Os tokens de acesso pessoal podem ser usados em várias contas da AWS. Diferentemente das GitHub integrações da Atlassian, não há nenhuma restrição GitLab para conectar a mesma conta a várias instâncias do AWS Security Agent.

Registrar uma GitLab conexão

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Escolha Adicionar integração.
3. Selecione GitLabe, em seguida, escolha Avançar.
4. Em Escolha um tipo de conta, selecione uma das seguintes opções:

- Pessoal - Conecte sua conta de GitLab usuário individual.
- Grupo - Conecte um GitLab grupo que contém vários projetos. Insira seu ID de grupo.

5. No campo Token de acesso, cole seu token de GitLab acesso.
6. No campo Nome do registro, insira um nome descritivo para essa conexão, como`Engineering-Team-GitLab`. Os caracteres válidos são letras, números, pontos, sublinhados e hífens.
7. Selecione Conectar.

   Você retorna à página Integrações, onde a nova conexão aparece com seu nome de registro.

Solucionar problemas de integração GitLab

Se você encontrar problemas durante o processo de GitLab integração, use as diretrizes a seguir para resolver problemas comuns.

Token inválido ou expirado

Sintomas

- A integração falha ao se conectar
- A integração que funcionava anteriormente deixa de funcionar
- Mensagem de erro indicando falha na autenticação

Resolução

1. Em GitLab, navegue até suas configurações de usuário e selecione Tokens de acesso.
2. Verifique se seu token não expirou.
3. Verifique se o token tem os escopos necessários para seu tipo.
4. Se o token tiver expirado, crie um novo token e atualize a integração no AWS Console.

Limitação de intervalo

Sintomas

- Falhas intermitentes durante a revisão do código
- Análise atrasada da solicitação de mesclagem

Resolução

- GitLab aplica limites de taxa às solicitações de API. Se você tiver um grande número de repositórios ou solicitações de mesclagem frequentes, algumas solicitações poderão ser limitadas.
- Aguarde até que a janela de limite de taxa seja redefinida ou entre em contato com o GitLab suporte para aumentar seus limites de tarifa.

Próximas etapas

Depois de se conectar GitLab ao AWS Security Agent:

- Navegue até o Espaço do Agente em que você deseja usar esses repositórios.
- Escolha Ativar revisão de código ou Configurar teste de penetração para conectar projetos específicos ao seu Espaço do Agente e configurar seu uso.
- Habilite a remediação de código para permitir que o AWS Security Agent envie solicitações de mesclagem com correções de vulnerabilidade
- Para GitLab instâncias hospedadas de forma privada, consulte the section called “Conecte o AWS Security Agent ao GitLab Self-Managed”

Conecte o AWS Security Agent ao GitLab Self-Managed

Conecte seu AWS Security Agent a uma GitLab Self-Managed instância para permitir recursos de análise de código, modelagem de ameaças, testes de penetração e remediação automatizada para repositórios hospedados em sua própria infraestrutura.

GitLab Self-Managed a integração funciona da mesma forma que o GitLab Cloud (consultethe section called “Conecte o AWS Security Agent aos GitLab repositórios”) com configuração adicional para conectividade de rede com sua instância privada. Antes de começar, analise the section called “Como as integrações funcionam com os Agent Spaces” para entender como um registro é reutilizado nos Agent Spaces e compartilhado entre os recursos.

> **Note**
>
> GitLab Self-Managed é registrado por meio da GitLabintegração, não de um tipo de integração separado. No fluxo de registro, você seleciona Usar endpoint GitLab auto-

hospedado e fornece o URL da sua instância. O GitLab fluxo hospedado na nuvem é descrito em. the section called “Conecte o AWS Security Agent aos GitLab repositórios”

Pré-requisitos

Antes de começar, verifique se você tem:

- Uma GitLab Self-Managed instância que é:
  - Acessível publicamente pela Internet, OU
  - Acessível por meio de uma conexão privada (consultethe section called “Conecte-se ao controle de origem hospedado de forma privada”)
- Um token de GitLab acesso com os escopos necessários para seu tipo de conexão:
  - Pessoal - Um token de acesso pessoal com todas as permissões de leitura e a `api` permissão.
  - Grupo - Um token de acesso de grupo com os `read_api` `read_repository` escopos e.
- Acesso de mantenedor ou proprietário aos projetos que você deseja conectar
- Sua GitLab instância deve fornecer tráfego HTTPS com uma versão mínima de TLS de 1.2

> **Note**
>
> Se sua GitLab Self-Managed instância usa certificados TLS emitidos por uma autoridade de certificação privada, você pode fornecer a chave PEM-encoded pública do certificado ao criar uma conexão privada. Isso permite que o AWS Security Agent confie na conexão TLS com sua instância.

Registrar uma GitLab Self-Managed conexão

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Escolha Adicionar integração.
3. Selecione GitLabe, em seguida, escolha Avançar.
4. Em Escolha um tipo de conta, selecione Pessoal ou Grupo. Se você selecionar Grupo, insira sua ID de grupo.
5. Selecione Usar endpoint GitLab auto-hospedado.

6. No campo URL do endpoint GitLab auto-hospedado, insira o URL da sua instância, por exemplo. `https://gitlab.example.com`
7. Se sua instância não estiver acessível publicamente, selecione Conectar ao endpoint usando uma conexão privada e, em seguida, escolha uma conexão privada existente ou crie uma nova. Consulte the section called “Conecte-se ao controle de origem hospedado de forma privada”.
8. No campo Token de acesso, cole seu token de GitLab acesso.
9. No campo Nome do registro, insira um nome descritivo para essa conexão. Os caracteres válidos são letras, números, pontos, sublinhados e hífens.
10. Selecione Conectar.

    Você retorna à página Integrações, onde a nova conexão aparece com seu nome de registro.

Conectividade privada

Se sua GitLab Self-Managed instância não estiver acessível publicamente, você deverá criar uma conexão privada antes de registrar a integração. Consulte the section called “Conecte-se ao controle de origem hospedado de forma privada” para obter instruções detalhadas.

> **Important**
>
> Service-managed conexões privadas exigem que a GitLab Self-Managed instância seja executada na mesma conta da AWS em que o Agent Space foi criado. Para acesso entre contas, use uma conexão privada autogerenciada na qual você fornece sua própria configuração de recursos do VPC Lattice.

Solucionar problemas de integração GitLab Self-Managed

Além das etapas de solução de problemasthe section called “Conecte o AWS Security Agent aos GitLab repositórios”, os seguintes problemas são específicos das instâncias autogerenciadas:

Instância inacessível

Sintomas

- Falha na conexão com tempo limite ou erro de rede
- A integração estava funcionando anteriormente, mas para de funcionar

Resolução

- Verifique se sua GitLab instância está em execução e acessível
- Se estiver usando uma conexão privada, verifique se o gateway de recursos do VPC Lattice está íntegro e se os ENIs têm conectividade de rede com sua instância
- Verifique se os grupos de segurança permitem tráfego na porta configurada
- Verifique se o certificado TLS é válido e não expirou

Erros do certificado TLS

Sintomas

- A conexão falha com SSL/TLS erro

Resolução

- Verifique se sua instância serve HTTPS com TLS 1.2 ou superior
- Se estiver usando uma autoridade de certificação privada, verifique se a chave PEM-encoded pública foi fornecida durante a configuração da conexão privada
- Verifique se o certificado não está expirado

Próximas etapas

Depois de se conectar GitLab Self-Managed ao AWS Security Agent:

- Navegue até o Espaço do Agente onde você deseja usar esses repositórios
- Escolha Ativar revisão de código ou Configurar teste de penetração para conectar projetos específicos
- Ative a correção de código para correções baseadas em solicitações de mesclagem

Remover uma GitLab integração

Remova uma GitLab integração quando você não precisar mais do AWS Security Agent para acessar repositórios de uma GitLab conta ou grupo específico.

Pré-requisitos para remoção

Antes de remover uma GitLab integração, verifique se você tem:

- Verificou quais Agent Spaces têm repositórios conectados a partir dessa integração
- Compreendi o impacto: a remoção dessa integração interromperá as conexões do repositório para análise de código, contexto de teste de penetração, modelagem de ameaças e remediação automatizada em todos os Agent Spaces usando repositórios dessa integração

Remova a integração do AWS Security Agent

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Localize a GitLab integração que você deseja remover.
3. Selecione a integração clicando nela.
4. Escolha Remover .
5. Revise a caixa de diálogo de confirmação e escolha Confirmar remoção.

> **Note**
>
> Depois de remover a integração, talvez você também queira revogar o Token de Acesso Pessoal GitLab para evitar mais acesso. Navegue até suas configurações de GitLab usuário e exclua ou revogue o token.

O mesmo processo se aplica às GitLab Self-Managed integrações.

Conecte o AWS Security Agent ao servidor GitHub corporativo

Conecte seu AWS Security Agent a uma instância do GitHub Enterprise Server (GHES) para permitir recursos de análise de código, modelagem de ameaças, testes de penetração e remediação automatizada para repositórios hospedados em sua própria infraestrutura.

GitHub A integração do Enterprise Server fornece os mesmos recursos que os hospedados na nuvem GitHub (consulte Conectar o AWS Security Agent aos GitHub repositórios) com configuração adicional para conectividade de rede com sua instância auto-hospedada. Antes de começar, analise the section called “Como as integrações funcionam com os Agent Spaces” para entender como um registro é reutilizado nos Agent Spaces e compartilhado entre os recursos.

> **Note**
>
> GitHub O Enterprise Server é registrado por meio da GitHubintegração, não de um tipo de integração separado. No fluxo de registro, você escolhe GitHub Enterprise Server como o tipo de instância. O GitHub.com fluxo hospedado na nuvem é descrito em. the section called "Conecte o AWS Security Agent aos GitHub repositórios"

Como funciona a integração do GitHub Enterprise Server

A análise do pull request acontece dentro do GitHub Enterprise Server. Depois de autorizar a conexão, conectar repositórios e habilitar comentários de revisão de código no Console de Gerenciamento da AWS, o AWS Security Agent verifica automaticamente as alterações em cada nova pull request (uma verificação diferencial apenas do código alterado) e publica as descobertas como comentários da pull request.

Você cria e executa análises completas de código — que examinam toda a base de código de um repositório — no aplicativo web do AWS Security Agent, não no GitHub Enterprise Server.

Os testes de penetração e a modelagem de ameaças são iniciados no aplicativo web do AWS Security Agent. Os usuários especificam domínios de destino e selecionam repositórios conectados para fornecer o contexto do aplicativo. Se você habilitar a remediação automatizada, os usuários poderão solicitar que o AWS Security Agent corrija as descobertas abrindo pull requests em repositórios conectados.

Pré-requisitos

Antes de começar, verifique se você tem:

- Uma instância do GitHub Enterprise Server que seja:
  - Acessível publicamente pela Internet, OU
  - Acessível por meio de uma conexão privada (consultethe section called "Conecte-se ao controle de origem hospedado de forma privada")
- Acesso de administrador do site ou administrador da organização em sua instância GHES
- Sua instância GHES deve fornecer tráfego HTTPS com uma versão TLS mínima de 1.2
- Permissões para configurar integrações no console de gerenciamento do AWS Security Agent

> **Note**
>
> GitHub As integrações do Enterprise Server podem ser usadas em várias contas da AWS.

Registrar uma conexão com o GitHub Enterprise Server

O registro de uma conexão com o GitHub Enterprise Server usa um fluxo de OAuth-based autorização.

> **Important**
>
> Conclua todas as etapas desse processo sem fechar o navegador ou sair da navegação. Se o processo de registro for interrompido, talvez seja necessário reiniciá-lo desde o início.

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Escolha Adicionar integração.
3. Selecione GitHube, em seguida, escolha Avançar.
4. Em Tipo de instância, selecione GitHub Enterprise Server.
5. No campo URL do servidor GitHub corporativo, insira a URL HTTPS da sua instância, por exemplo`https://github.example.com`.
6. Se sua instância não estiver acessível publicamente, selecione Conectar ao endpoint usando uma conexão privada e, em seguida, escolha uma conexão privada existente ou crie uma nova. Consulte the section called “Conecte-se ao controle de origem hospedado de forma privada”.
7. Na seção Detalhes do registro, configure os seguintes campos:
   a. Nome do registro - insira um nome descritivo para essa conexão. Os caracteres válidos são letras, números, pontos, sublinhados e hífens.
   b. GitHub tipo de conta - Selecione Organização ou Usuário.
   c. Nome da organização (aparece somente se você selecionou Organização) - Insira o nome exato da sua organização do GitHub Enterprise Server. Os nomes diferenciam letras maiúsculas de minúsculas.
8. Selecione Conectar.

> **Note**
>
> O AWS Security Agent redireciona você para fora do console para concluir a autorização com sua instância do GitHub Enterprise Server. Após a conclusão da autorização, você retorna ao console e a nova conexão aparece na página Integrações.

Conectividade privada

Se sua instância do GitHub Enterprise Server não estiver acessível publicamente, você deverá criar uma conexão privada antes de registrar a integração. Consulte the section called “Conecte-se ao controle de origem hospedado de forma privada” para obter instruções detalhadas.

> **Important**
>
> Service-managed conexões privadas exigem que a instância GHES seja executada na mesma conta da AWS em que o Agent Space é criado. Para acesso entre contas, use uma conexão privada autogerenciada.

> **Note**
>
> Se sua instância GHES usa certificados TLS emitidos por uma autoridade de certificação privada, forneça a chave PEM-encoded pública do certificado ao criar a conexão privada.

Solucionar problemas de integração com o GitHub Enterprise Server

Se você encontrar problemas ao conectar o AWS Security Agent ao GitHub Enterprise Server, use as orientações a seguir para diagnosticar e resolver problemas comuns.

Falha de redirecionamento do OAuth

Sintomas

- Os redirecionamentos do navegador falham durante o fluxo de autorização
- Página de erro exibida após a autorização no GHES

Resolução

- Verifique se sua instância GHES está acessível a partir do seu navegador
- Verifique se o URL de retorno de chamada do OAuth está configurado corretamente
- Reinicie o processo de integração desde o início

Instância inacessível

Sintomas

- Falha na conexão com tempo limite ou erro de rede

Resolução

- Verifique se sua instância GHES está em execução e acessível
- Se estiver usando uma conexão privada, verifique a conectividade do VPC Lattice
- Verifique se os grupos de segurança permitem tráfego na porta configurada
- Verifique se o certificado TLS é válido (mínimo TLS 1.2)

Próximas etapas

Depois de conectar o GitHub Enterprise Server ao AWS Security Agent:

- Navegue até o Espaço do Agente onde você deseja usar esses repositórios
- Escolha Habilitar revisão de código ou Configurar teste de penetração para conectar repositórios específicos
- Habilite a remediação de código para permitir que o AWS Security Agent envie pull requests com correções de vulnerabilidade

Remover uma integração com o GitHub Enterprise Server

Remova uma integração do GitHub Enterprise Server quando você não precisar mais do AWS Security Agent para acessar repositórios de uma instância específica do GHES.

Pré-requisitos para remoção

Antes de remover uma integração do GHES:

- Verifique quais Agent Spaces têm repositórios conectados a partir dessa integração
- Entenda o impacto: a remoção interromperá a revisão de código, o contexto do teste de penetração, a modelagem de ameaças e a remediação automatizada de todos os repositórios conectados

Remova a integração

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Localize a integração do GitHub Enterprise Server que você deseja remover.
3. Selecione a integração.
4. Escolha Remover .
5. Revise a caixa de diálogo de confirmação e escolha Confirmar remoção.

> **Note**
>
> Após a remoção, talvez você também queira revogar o aplicativo OAuth na sua instância GHES. Navegue até as configurações da sua organização GHES e remova o aplicativo autorizado.

Encontre seu ID de instalação da Atlassian

Antes de registrar uma integração do Bitbucket ou do Confluence no console da AWS, você precisa encontrar o ID de instalação do aplicativo AWS Security Agent Forge instalado no seu site da Atlassian. Você cola esse ID no fluxo de registro.

Para Bitbucket

1. No Bitbucket, navegue até as configurações do seu espaço de trabalho.
2. Selecione Forge Apps na barra lateral.
3. Localize Connect with AWS Security Agent na lista de aplicativos instalados.
4. Escolha Copiar para a área de transferência para copiar o ID de instalação.

Para Confluence

1. No Confluence, navegue até Administração do Confluence.
2. Selecione Aplicativos na barra lateral.
3. Localize Connect with AWS Security Agent na lista de aplicativos instalados.
4. Escolha Copiar para a área de transferência para copiar o ID de instalação.

Você precisará desse ID de instalação ao concluir o registro no console de gerenciamento do AWS Security Agent.

Conecte o AWS Security Agent aos repositórios do Bitbucket

Conecte seu AWS Security Agent aos repositórios do Bitbucket Cloud para habilitar recursos de análise de código, modelagem de ameaças, testes de penetração e remediação automatizada. Antes de começar, analise the section called “Como as integrações funcionam com os Agent Spaces” para entender como um registro é reutilizado nos Agent Spaces e compartilhado entre os recursos.

A integração com o Bitbucket serve a vários propósitos:

- Revisão de código - Analise automaticamente as alterações de código em cada pull request de acordo com seus requisitos de segurança organizacional e execute varreduras de repositório completo sob demanda
- Modelagem de ameaças - Forneça compreensão do aplicativo analisando o código-fonte, os fluxos de dados e a arquitetura
- Contexto do teste de penetração - Forneça compreensão do aplicativo para testes de penetração
- Remediação automatizada - envie pull requests com correções para vulnerabilidades descobertas durante avaliações de segurança

Conectar o Bitbucket ao AWS Security Agent requer a instalação do aplicativo AWS Security Agent Forge em seu site da Atlassian e a conclusão do fluxo de autorização do OAuth.

> **Note**
>
> O AWS Security Agent é compatível somente com o Bitbucket Cloud. O Bitbucket Server e o Bitbucket Data Center não são suportados.

Como funciona a integração com o Bitbucket

A análise do pull request acontece dentro do Bitbucket. Depois de instalar o aplicativo Forge, conectar repositórios e ativar os comentários de revisão de código no AWS Management Console, o AWS Security Agent verifica automaticamente as alterações em cada nova pull request (uma verificação diferencial apenas do código alterado) e publica as descobertas como comentários da pull request.

Você cria e executa análises completas de código — que examinam toda a base de código de um repositório — no aplicativo web do AWS Security Agent, não no Bitbucket.

Os testes de penetração e a modelagem de ameaças são iniciados no aplicativo web do AWS Security Agent. Os usuários especificam domínios de destino e selecionam repositórios conectados para fornecer o contexto do aplicativo.

> **Note**
>
> A correção automatizada não está disponível para repositórios públicos do Bitbucket para evitar a divulgação de vulnerabilidades antes que elas sejam corrigidas.

Pré-requisitos

Antes de começar, verifique se você tem:

- Um espaço de trabalho do Bitbucket Cloud com acesso de administrador
- Uma conta Atlassian com privilégios de administrador do site
- Permissões para configurar integrações no console de gerenciamento do AWS Security Agent

> **Important**
>
> Um site da Atlassian só pode ser associado a uma conta da AWS por região. Se você precisar conectar o mesmo site do Bitbucket ao AWS Security Agent em uma conta da AWS diferente na mesma região, primeiro remova a integração existente.

> **Note**
>
> Os preços do aplicativo Atlassian Forge se aplicam a essa integração. Para obter mais informações, consulte os preços da plataforma Forge na documentação da Atlassian.

Registrar uma conexão do Bitbucket

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Escolha Adicionar integração.
3. Selecione Bitbucket e, em seguida, escolha Avançar.
4. Instale o aplicativo AWS Security Agent Forge em seu espaço de trabalho do Bitbucket, seguindo as instruções na tela. Após a instalação, copie o ID da instalação. Consulte the section called "Encontre seu ID de instalação da Atlassian".
5. No campo ID de instalação, cole o ID de instalação que você copiou do Bitbucket.
6. No campo Espaço de trabalho do Bitbucket, insira o slug do espaço de trabalho do Bitbucket, por exemplo. `acme-corp`
7. Escolha Authorize.

   Você será redirecionado para a Atlassian para autorizar o AWS Security Agent a acessar seu espaço de trabalho do Bitbucket. Depois que a autorização for concluída, você retornará ao console.
8. Na seção Detalhes do registro, insira um nome de registro para essa conexão. Os caracteres válidos são letras, números, pontos, sublinhados e hífens.
9. Selecione Conectar.

> **Note**
>
> Atraso no provisionamento após a conexão
>
> Depois de escolher Connect, o provisionamento no lado da Atlassian pode levar alguns minutos. Durante esse período, a integração relata um status pendente. Se você tentar selecionar repositórios ou habilitar a revisão de código antes da conclusão do provisionamento, talvez veja uma mensagem informando que a instalação ainda não está disponível. Aguarde alguns minutos e tente novamente.

Solucionar problemas de integração com o Bitbucket

Se você encontrar problemas durante o processo de integração do Bitbucket, use as orientações a seguir para resolver problemas comuns.

Não foi possível concluir o registro

Se o processo de registro for interrompido (navegador fechado, tempo limite da sessão), o aplicativo Forge poderá ser instalado em seu site da Atlassian, mas não registrado no console da AWS.

Resolução

- Retorne ao console do AWS Security Agent e reinicie o processo de integração.
- O aplicativo Forge permanece instalado e não precisa ser reinstalado.

Site já conectado a outra conta da AWS

Sintomas

- Erro indicando que o site da Atlassian já está associado a outra conta

Resolução

- Um site da Atlassian só pode ser conectado a uma conta da AWS por região.
- Identifique qual conta da AWS tem a integração existente e use essa conta, ou remova primeiro a integração existente.

A instalação não está disponível

Sintomas

- Quando você seleciona repositórios ou ativa a revisão de código, você vê uma mensagem informando que a instalação do Bitbucket está no status`PENDING_INSTALLATION`, não no status. `AVAILABLE`

Resolução

- A instalação ainda está sendo provisionada no lado da Atlassian. O provisionamento normalmente é concluído em alguns minutos. Aguarde alguns minutos e tente novamente.

- Se o status não ficar disponível após alguns minutos, remova a integração e registre-a novamente. Antes de se registrar novamente, desinstale o aplicativo Forge do seu espaço de trabalho do Bitbucket. Para obter instruções, consulte Remover uma integração com o Bitbucket. Se você se registrar novamente sem desinstalar o aplicativo Forge anterior, a instalação pode ficar paralisada em um estado pendente.

Próximas etapas

Depois de conectar o Bitbucket ao AWS Security Agent:

- Navegue até o Espaço do Agente onde você deseja usar esses repositórios
- Escolha Habilitar revisão de código ou Configurar teste de penetração para conectar repositórios específicos ao seu Agent Space
- Habilite a remediação de código para permitir que o AWS Security Agent envie pull requests com correções de vulnerabilidade

Remover uma integração com o Bitbucket

Remova uma integração do Bitbucket quando você não precisar mais do AWS Security Agent para acessar repositórios de um espaço de trabalho específico do Bitbucket.

Pré-requisitos para remoção

Antes de remover uma integração do Bitbucket:

- Verifique quais Agent Spaces têm repositórios conectados a partir dessa integração
- Entenda o impacto: a remoção interromperá a revisão de código, o contexto do teste de penetração, a modelagem de ameaças e a remediação automatizada de todos os repositórios conectados

Etapa 1: remover a integração do AWS Security Agent

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Localize a integração do Bitbucket que você deseja remover.
3. Selecione a integração.
4. Escolha Remover .

5. Revise a caixa de diálogo de confirmação e escolha Confirmar remoção.

Etapa 2: desinstalar o aplicativo Forge

Depois de remover a integração do AWS Security Agent, desinstale o aplicativo Forge do seu site da Atlassian:

1. No Bitbucket, navegue até as configurações do espaço de trabalho.
2. Selecione Forge Apps.
3. Localize Connect with AWS Security Agent.
4. Clique em Desinstalar.

> **Important**
>
> O aplicativo Forge não é desinstalado automaticamente
>
> Remover a integração no console do AWS Security Agent não desinstala o aplicativo Forge do seu site da Atlassian. Se você planeja registrar o mesmo espaço de trabalho do Bitbucket novamente, primeiro desinstale o aplicativo Forge. Caso contrário, a nova instalação pode ficar presa em um estado pendente.

Conecte o AWS Security Agent ao Confluence

Conecte seu AWS Security Agent ao Confluence Cloud para fornecer contexto de documentação para avaliações de segurança. Ao contrário dos provedores de código, o Confluence serve como uma fonte de documentação que fornece modelos de ameaças, documentos de arquitetura, especificações de API e outros materiais que aprimoram a qualidade das análises de segurança. Antes de começar, analise the section called “Como as integrações funcionam com os Agent Spaces” para entender como um registro é reutilizado nos Agent Spaces.

A integração do Confluence serve a vários propósitos:

- Contexto de revisão de design - Forneça documentos arquitetônicos e especificações de design para análises de design de segurança
- Modelagem de ameaças - Forneça modelos de ameaças existentes e documentação do sistema para análise de ameaças

- Contexto do teste de penetração - Forneça documentação do aplicativo para uma compreensão mais profunda durante o teste de penetração

Conectar o Confluence ao AWS Security Agent requer a instalação do aplicativo AWS Security Agent Forge em seu site da Atlassian e a conclusão do fluxo de autorização do OAuth.

> **Note**
>
> O AWS Security Agent é compatível somente com o Confluence Cloud. O Confluence Data Center e o Confluence Server não são suportados.

Como funciona a integração com o Confluence

O Confluence é um provedor de documentação em vez de um provedor de código-fonte. Depois de instalar o aplicativo Forge e conectar espaços ou páginas no AWS Management Console, o AWS Security Agent pode acessar seu conteúdo do Confluence para fornecer contexto durante as avaliações de segurança.

O AWS Security Agent lê o conteúdo da página para entender a arquitetura do seu aplicativo, os requisitos de segurança e as decisões de design. Esse contexto melhora a qualidade e a relevância das descobertas de segurança durante análises de design, análises de código e testes de penetração.

Pré-requisitos

Antes de começar, verifique se você tem:

- Um site do Confluence Cloud com acesso de administrador
- Uma conta Atlassian com privilégios de administrador do site
- Permissões para configurar integrações no console de gerenciamento do AWS Security Agent

> **Important**
>
> Um site da Atlassian só pode ser associado a uma conta da AWS por região. Se você precisar conectar o mesmo site do Confluence ao AWS Security Agent em uma conta da AWS diferente na mesma região, primeiro remova a integração existente.

> **Note**
>
> Os preços do aplicativo Atlassian Forge se aplicam a essa integração. Para obter mais informações, consulte os preços da plataforma Forge na documentação da Atlassian.

Registrar uma conexão do Confluence

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Escolha Adicionar integração.
3. Selecione Confluence e escolha Avançar.
4. Instale o aplicativo AWS Security Agent Forge em seu site da Atlassian, seguindo as instruções na tela. Após a instalação, copie o ID da instalação. Consulte the section called “Encontre seu ID de instalação da Atlassian”.
5. No campo URL do site do Confluence, insira a URL do seu site, por exemplo. `https://acme.atlassian.net`
6. No campo ID de instalação, cole o ID de instalação que você copiou do Confluence.
7. Escolha Authorize.

   Você será redirecionado para a Atlassian para autorizar o AWS Security Agent a acessar seu site do Confluence. Depois que a autorização for concluída, você retornará ao console.
8. Na seção Detalhes do registro, insira um nome de registro para essa conexão. Os caracteres válidos são letras, números, pontos, sublinhados e hífens.
9. Selecione Conectar.

Selecionar páginas para um Espaço do Agente

Depois de registrar a integração do Confluence, conecte páginas específicas a um Agent Space. Selecionar uma página concede ao AWS Security Agent acesso de leitura (busca) ao conteúdo dessa página. Não há opções de capacidade por página — o agente lê todas as páginas conectadas. Na etapa de revisão do assistente de conexão, você pode remover todas as páginas que não deseja que o agente acesse.

Solucionar problemas de integração com o Confluence

Se você encontrar problemas durante o processo de integração do Confluence, use as diretrizes a seguir para resolver problemas comuns.

Não foi possível concluir o registro

Se o processo de registro for interrompido, o aplicativo Forge poderá ser instalado em seu site da Atlassian, mas não registrado no AWS Console.

Resolução

- Retorne ao console do AWS Security Agent e reinicie o processo de integração.
- O aplicativo Forge permanece instalado e não precisa ser reinstalado.

Aplicativo Forge desinstalado da Atlassian

Se o aplicativo AWS Security Agent Forge for desinstalado do seu site da Atlassian enquanto a integração ainda existir no AWS Security Agent:

Sintomas

- A integração aparece no console da AWS, mas não pode acessar o conteúdo do Confluence
- Erros ao tentar listar ou ler páginas

Resolução

- Reinstale o aplicativo Forge no Atlassian Marketplace
- Se o problema persistir, remova a integração no AWS Console e registre-se novamente

Site já conectado a outra conta da AWS

Resolução

- Um site da Atlassian só pode ser conectado a uma conta da AWS por região.
- Identifique qual conta da AWS tem a integração existente e use essa conta, ou remova primeiro a integração existente.

Próximas etapas

Depois de conectar o Confluence ao AWS Security Agent:

- Navegue até o Espaço do Agente onde você deseja usar esta documentação.
- Selecione páginas específicas para incluir como contexto para análises de design e testes de penetração
- Faça upload de documentação adicional via S3, se necessário (consultethe section called "Forneça recursos de agente a partir de um bucket S3")

Remover uma integração com o Confluence

Remova uma integração do Confluence quando você não precisar mais do AWS Security Agent para acessar a documentação de um site específico do Confluence.

Pré-requisitos para remoção

Antes de remover uma integração com o Confluence:

- Verifique quais Agent Spaces têm páginas conectadas a partir dessa integração
- Entenda o impacto: a remoção quebrará o contexto da documentação para análises de design, testes de penetração e modelagem de ameaças em todos os Agent Spaces usando o conteúdo dessa integração

Etapa 1: remover a integração do AWS Security Agent

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Localize a integração do Confluence que você deseja remover.
3. Selecione a integração.
4. Escolha Remover .
5. Revise a caixa de diálogo de confirmação e escolha Confirmar remoção.

Etapa 2: desinstalar o aplicativo Forge (opcional)

Depois de remover a integração do AWS Security Agent, você pode, opcionalmente, desinstalar o aplicativo Forge do seu site da Atlassian:

1. No Confluence, navegue até Administração do Confluence.
2. Selecione Aplicativos.
3. Localize Connect with AWS Security Agent.
4. Clique em Desinstalar.

Conecte-se ao controle de origem hospedado de forma privada

> **Important**
>
> As conexões privadas estão na versão prévia do AWS Security Agent e estão sujeitas a alterações.

O AWS Security Agent pode se conectar a sistemas de controle de origem executados em redes privadas, como GitLab Self-Managed instâncias e servidores GitHub corporativos. As conexões privadas usam o Amazon VPC Lattice para estabelecer conectividade segura entre o AWS Security Agent e sua infraestrutura privada sem expor seus sistemas à Internet pública.

Como as conexões privadas funcionam

Uma conexão privada cria um caminho de rede seguro entre o AWS Security Agent e um recurso de destino em sua VPC. Nos bastidores, o AWS Security Agent usa o Amazon VPC Lattice para estabelecer essa conectividade. O VPC Lattice é um serviço de rede de aplicativos da AWS para conectar, proteger e monitorar o tráfego entre serviços em VPCs e contas, sem que você gerencie a infraestrutura de rede subjacente.

Quando você cria uma conexão privada:

1. Você fornece a VPC, as sub-redes e (opcionalmente) os grupos de segurança que têm conectividade de rede com seu serviço de destino.
2. O AWS Security Agent cria um gateway de recursos gerenciado por serviços e provisiona interfaces de rede elásticas (ENIs) nas sub-redes que você especificou.
3. O agente usa o gateway de recursos para rotear o tráfego para o endereço IP ou nome DNS do serviço de destino pelo caminho da rede privada.

Service-managed versus conexões privadas autogerenciadas

O AWS Security Agent oferece suporte a dois modos para conexões privadas:

Service-managed(recomendado para configurações simples):

- O AWS Security Agent cria e gerencia o gateway de recursos VPC Lattice para você.
- O serviço de destino deve estar em execução na mesma conta da AWS em que o Espaço do Agente foi criado.
- Não é possível abranger várias contas da AWS.

Self-managed(para configurações avançadas ou de várias contas):

- Você cria e gerencia sua própria configuração de recursos do VPC Lattice.
- Você fornece o Amazon Resource Name (ARN) da sua configuração de recurso existente.
- Oferece suporte ao acesso entre contas (o cliente gerencia a rede entre contas).

Pré-requisitos

Antes de criar uma conexão privada, verifique se você tem:

- Um Espaço de Agentes ativo
- Um serviço de destino de acesso privado (GitLab Self-Managed ou GitHub Enterprise Server) em um endereço IP privado conhecido ou nome DNS
- O serviço de destino deve fornecer tráfego HTTPS com uma versão TLS mínima de 1.2
- Sub-redes em sua VPC (1-20 sub-redes). Recomendamos selecionar sub-redes em várias zonas de disponibilidade para alta disponibilidade.
- (Opcional) Grupos de segurança para controlar o tráfego para as ENIs (até 5)

> **Note**
>
> As seguintes zonas de disponibilidade não são compatíveis com o VPC Lattice:`use1-az3`,,,`usw1-az2`,`apne1-az3`,, `apne2-az2euc1-az2`,`euw1-az4`. `cac1-az3` `ilc1-az2`

> **Note**
>
> Conexões privadas são recursos em nível de conta. Depois de criar uma conexão privada, você pode reutilizá-la em várias integrações e espaços de agentes que precisam alcançar o mesmo host.

> **Note**
>
> Quando você seleciona uma conexão privada para um provedor que usa a autenticação OAuth, a conexão privada se aplica tanto ao endpoint do provedor quanto ao endpoint da troca de tokens. Certifique-se de que a conexão privada esteja configurada com um endereço de host que possa rotear o tráfego para os dois endpoints.

Crie uma conexão privada

1. No console de gerenciamento do AWS Security Agent, navegue até Integrações.
2. Escolha a guia Conexões privadas.
3. Escolha Criar conexão privada.
4. Em Detalhes da conexão, insira um Nome. Os nomes podem conter letras minúsculas, números e hífens e devem começar e terminar com uma letra ou número.
5. Na seção Detalhes da conexão, escolha como o AWS Security Agent se conecta ao seu serviço de destino:
   - Para que o AWS Security Agent crie e gerencie a conexão, deixe a opção Usar a configuração de recursos existentes desmarcada e, em seguida, preencha as seções Localização do recurso, Controle de acesso e Detalhes da meta do serviço.
   - Para rotear por meio de uma configuração de recursos do VPC Lattice que você já criou, selecione Usar configuração de recursos existente e conclua a seção Configuração de recursos existentes. As seções gerenciadas por serviços não se aplicam.
6. (Service-managed) Em Localização do recurso:
   a. Selecione a VPC em que seu recurso está localizado.
   b. Selecione uma ou mais sub-redes. Recomendamos selecionar sub-redes em pelo menos duas zonas de disponibilidade.
   c. (Opcional) Selecione um tipo de endereço IP (IPv4, IPv6 ou pilha dupla).

7. (Service-managed) Sob controle de acesso:
   a. Selecione os grupos de segurança associados aos seus recursos conectados.
   b. (Opcional) Em Configuração avançada, insira Intervalos de portas — até 11 portas ou intervalos TCP separados por vírgula, por exemplo. `443, 8080-8090`
8. (Service-managed) Em Detalhes do alvo do serviço:
   a. No campo Endereço do host, insira o nome do host DNS ou o endereço IP do seu serviço de destino.
   b. Para resolução de DNS, selecione Público para um endereço de host que possa ser resolvido publicamente ou Em VPC (DNS privado) para um endereço que pode ser resolvido somente dentro da VPC.
   c. (Opcional) No campo Chave pública do certificado, insira a chave PEM-encoded pública se o destino usar um certificado autoassinado ou uma autoridade de certificação privada. Isso permite que o AWS Security Agent confie na conexão TLS.
9. (Self-managed) Em Configuração de recursos existentes, em Configuração de recursos, selecione uma configuração de recurso VPC Lattice na lista ou insira uma ID de configuração do recurso (começando com `rcfg-`) ou seu ARN. A lista mostra as configurações de recursos na região atual.
10. Escolha Criar conexão.

    O status da conexão muda para Criar em andamento. Esse processo pode levar até 10 minutos. Quando concluído, o status muda para Disponível.

Use uma conexão privada com uma integração

Ao registrar uma GitLab Self-Managed integração com o GitHub Enterprise Server, selecione Conectar ao endpoint usando uma conexão privada e, em seguida, escolha sua conexão na lista de conexões privadas. O AWS Security Agent encaminha o tráfego para o provedor por meio dessa conexão. Somente conexões com status Disponível aparecem na lista.

Você também pode escolher Criar uma conexão privada durante o registro. O AWS Security Agent preserva seu rascunho de registro enquanto você cria a conexão e, em seguida, retorna ao fluxo de registro.

Segurança

- Sem exposição pública à Internet — Todo o tráfego permanece na rede da AWS.

- Customer-controlled grupos de segurança - Você gerencia as regras de tráfego de entrada e saída.
- Service-linked função com menor privilégio — O AWS Security Agent usa uma função vinculada a serviços com escopo definido para recursos marcados com. `AWSSecurityAgentManaged`

> **Note**
>
> Se sua organização tem políticas de controle de serviço (SCPs) que restringem as ações da API VPC Lattice, o gateway de recursos gerenciados por serviços é criado por meio de uma função vinculada ao serviço. Certifique-se de que seus SCPs permitam as ações necessárias para a função vinculada ao serviço.

Verificar uma conexão privada

Depois que a conexão privada atingir o status Disponível, verifique a conectividade:

- Certifique-se de que seu serviço de destino esteja funcionando e aceitando conexões na porta esperada
- Verifique se os grupos de segurança conectados às ENIs permitem tráfego de saída na porta de destino
- Verifique se o grupo de segurança do seu serviço permite tráfego de entrada dos IPs do plano de dados VPC Lattice dentro do intervalo CIDR da VPC
- Confirme se as tabelas de rotas de sub-rede permitem tráfego entre as sub-redes ENI e seu serviço

Excluir uma conexão privada

1. No console do AWS Security Agent, navegue até Integrações.
2. Escolha a guia Conexões privadas.
3. Selecione a conexão privada que você deseja excluir.
4. Escolha Excluir.

> **Important**
>
> Você não pode excluir uma conexão privada que esteja sendo usada atualmente por uma integração. Primeiro remova a integração e, em seguida, exclua a conexão privada.

Próximas etapas

- the section called “Conecte o AWS Security Agent ao GitLab Self-Managed”- Conecte uma GitLab instância privada
- the section called “Conecte o AWS Security Agent ao servidor GitHub corporativo”- Conecte um servidor GitHub corporativo privado

Conecte o agente aos recursos privados da VPC

Se o aplicativo no qual você deseja executar um teste de penetração não estiver disponível na Internet pública, você precisará fornecer ao AWS Security Agent uma configuração de VPC. O AWS Security Agent usará essa configuração de VPC, incluindo uma VPC, sub-rede e grupos de segurança, para acessar o aplicativo.

> **Note**
>
> Ao testar endpoints em uma VPC privada, somente endpoints resolvidos para IPs em intervalos de IP privados conhecidos são permitidos (consulte Blocos CIDR de VPC para obter mais informações). Os seguintes intervalos de IPv4 e IPv6 são permitidos:
>
> ```
> 10.0.0.0/8
> 172.16.0.0/12
> 192.168.0.0/16
> fd00::/8
> ```

> **Note**
>
> Ao se conectar a uma sub-rede, o AWS Security Agent criará uma ENI (Elastic Network Interface) na sub-rede configurada para o teste de penetração. Essa ENI não tem um endereço IP público associado, o que significa que ela não pode se comunicar com os VPC

Internet Gateways em sub-redes públicas. Se seu teste de penetração exigir acesso aberto à Internet, use uma sub-rede privada com um VPC NAT Gateway associado

Você concede ao AWS Security Agent acesso geral a uma VPC a partir do AWS Management Console. No aplicativo web do Security Agent, os usuários selecionam a configuração específica para um teste de penetração.

Para adicionar uma VPC no Espaço do Agente

1. Navegue até a página de visão geral do Agent Space
2. Selecione Ações e, em seguida, Editar configuração do teste de penetração
3. Sob o título VPC, especifique a VPC, as sub-redes e os grupos de segurança

Você pode adicionar até 5 VPCs.

Permissões necessárias da função de serviço do espaço do agente

Para executar um teste de penetração com uma VPC, sua função de serviço de espaço de agente deve incluir as seguintes permissões:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DescribeDhcpOptions",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeNetworkInterfaceAttribute",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeVpcs",
        "ec2:ModifyNetworkInterfaceAttribute",
        "ec2:DeleteNetworkInterface"
      ],
      "Resource": "*"
    },
    {
```

```
      "Effect": "Allow",
      "Action": "ec2:CreateNetworkInterfacePermission",
      "Resource": "arn:aws:ec2:{{region}}:{{accountId}}:network-interface/*",
      "Condition": {
        "ArnEquals": {
          "ec2:Subnet": [
            "arn:aws:ec2:{{region}}:{{accountId}}:subnet/[[subnetIds]]"
          ]
        }
      }
    }
  ]
}
```

Para selecionar uma configuração de VPC específica para um teste de penetração no aplicativo web do Security Agent

1. Navegue até a página de visão geral dos testes de penetração
2. Selecione o teste de penetração ao qual você precisa adicionar a configuração de VPC e, em seguida, escolha Modificar detalhes do pentest
3. Selecione Avançar para acessar a seção Recursos de VPC
4. Selecione os grupos VPC, sub-rede e segurança
5. Selecione Avançar para chegar à última seção e salvar o teste de penetração

> **Note**
>
> Cross-account Atualmente, o teste de penetração é suportado para recursos de VPC (sub-redes e grupos de segurança) compartilhados usando o AWS Resource Access Manager. Os segredos do Secrets Manager e as funções do Lambda usados para credenciais de autenticação devem ser configurados na mesma conta da AWS que a configuração do AWS Security Agent.

Executando um teste de penetração em recursos de VPC em outra conta da AWS

Você pode executar testes de penetração em recursos de VPC compartilhados com sua conta usando o AWS Resource Access Manager. Ambas as contas devem fazer parte da mesma organização da AWS.

1. (Opcional) Habilite o compartilhamento automático de recursos para sua organização da AWS

```
aws ram enable-sharing-with-aws-organization
```

2. Usando credenciais da conta da AWS que possui os recursos da VPC, compartilhe os recursos da sub-rede e do grupo de segurança com a conta do proprietário do teste de penetração

```
aws ram create-resource-share \
    --name SharePentestResources \
    --resource-arns <subnet ARN> <security group ARN> \
    --principals <penetration test owner account ID>
```

3. Navegue até a página de visão geral do Agent Space
4. Selecione Teste de penetração e localize o nome da função de serviço
5. Verifique se a função do IAM concede acesso aos recursos compartilhados da VPC
6. Selecione Ações e, em seguida, Editar configuração do teste de penetração
7. Sob o título VPC, especifique a VPC, as sub-redes e os grupos de segurança compartilhados e salve a configuração atualizada.
8. Navegue até a página de visão geral dos testes de penetração no aplicativo web AWS Security Agent
9. Selecione o teste de penetração ao qual você precisa adicionar a configuração de VPC e, em seguida, escolha Modificar detalhes do pentest
10. Atualize o teste de penetração para usar os recursos compartilhados da VPC

Configurar recursos

Ative e configure testes de penetração, análise de código e modelagem de ameaças para seus Agent Spaces, incluindo remediação, fontes do S3 e domínios de destino.

Ativar teste de penetração

Configure o AWS Security Agent para executar testes de penetração autônomos em seus aplicativos. Essa configuração permite que o AWS Security Agent acesse seus recursos da AWS, verifique a

propriedade do domínio e realize testes de segurança abrangentes que identificam vulnerabilidades exploráveis em seus aplicativos web e APIs.

A ativação do teste de penetração permite que o AWS Security Agent valide continuamente a segurança do seu aplicativo sem os atrasos dos testes manuais. Ao configurar as integrações necessárias da AWS, você garante que o AWS Security Agent possa testar aplicativos públicos e privados, registrar descobertas e acessar credenciais para testes autenticados. CloudWatch

Neste procedimento, você configurará domínios de destino, configurará opcionalmente VPC, CloudWatch registro, armazenamento de credenciais e funções Lambda para testes, configurará integrações do S3 para fornecer contexto adicional e configurará o acesso ao serviço por meio de funções do IAM.

Pré-requisitos

Antes de começar, verifique se você tem:

- Conta da AWS com permissões administrativas para criar funções do IAM
- Capacidade de verificação da propriedade do domínio (modificação do registro DNS ou HTTP)
- Domínios ou aplicativos de destino para testar
- (Opcional) Detalhes da configuração da VPC se estiver testando aplicativos privados
- (Opcional) Bucket S3 se fornecer artefatos adicionais ao AWS Security Agent

Etapa 1: configurar o domínio

Na primeira etapa do assistente, especifique os domínios de destino que você deseja testar e como o AWS Security Agent deve verificar a propriedade.

1. Na seção Domínios de destino, insira seu domínio no campo Domínio.
2. Selecione um método de verificação:
   - DNS_TXT — Prove a propriedade do domínio adicionando um registro TXT à configuração de DNS do seu domínio.
   - HTTP_ROUTE — Prove a propriedade do domínio hospedando um arquivo de verificação em uma URL específica em seu domínio.
   - PRIVATE_VPC - Só pode ser usado para testes de penetração de VPC privados. Verifica se o domínio é resolvido para um IP em um intervalo CIDR privado

- Para obter mais informações, consulte the section called “Habilite um domínio de aplicativo para testes de penetração”.

3. Escolha Adicionar outro domínio para adicionar outros domínios (até 5 no total).
4. Escolha Avançar para prosseguir com a verificação do domínio.

Etapa 2: verificar domínios

Na segunda etapa do assistente, verifique a propriedade de cada domínio que você configurou. O AWS Security Agent exige propriedade verificada antes de realizar testes de penetração.

1. Analise os domínios listados na tabela Domínios de destino.
2. Para cada domínio, selecione-o e acione a verificação com base no método escolhido:
   - Domínios do Route 53 (mesma conta da AWS): escolha a One-click verificação. O AWS Security Agent cria automaticamente o registro DNS e conclui a verificação.
   - DNS TXT (outros provedores de DNS): copie o token de verificação, adicione o registro TXT ao seu registrador de DNS, selecione o domínio e escolha Verificar.
   - Rota HTTP: coloque o token de verificação no caminho de rota necessário em seu servidor web, selecione o domínio e escolha Verificar. Para obter detalhes, consulte the section called “Habilite um domínio de aplicativo para testes de penetração”.
3. Escolha Avançar para prosseguir com a configuração opcional.

> **Note**
>
> Sub-domains de um domínio verificado não exigem verificação individual ao usar a verificação de rota DNS, TXT ou HTTP. Para a verificação de VPC privada, o nome de domínio do endpoint de destino deve corresponder ao nome de domínio completo configurado para verificação.

> **Note**
>
> Para domínios privados dentro de uma VPC, você pode continuar mesmo que o status de verificação do domínio seja INACESSÍVEL. O AWS Security Agent tentará a verificação de domínio para endpoints privados no início de cada execução de pentest.

Etapa 3: (opcional) configurar recursos adicionais

A terceira etapa do assistente permite que você configure recursos opcionais da AWS para expandir seu escopo de testes de penetração. Todas as seções desta etapa são opcionais e estão reduzidas por padrão.

(Opcional) Definir as configurações de VPC

Se você planeja testar domínios de destino privados hospedados em uma VPC, defina as configurações da VPC para o AWS Security Agent. Essa seção é opcional e é reduzida por padrão.

1. Expanda a seção VPCs.
2. No menu suspenso da VPC, selecione a VPC que hospeda seus domínios de destino privados.
3. No menu suspenso Sub-rede, selecione as sub-redes VPC que o AWS Security Agent deve usar:

> **Tip**
>
> Para alta disponibilidade, selecione várias sub-redes de várias zonas de disponibilidade. Certifique-se de que suas sub-redes incluam um gateway NAT para conectividade de saída.

4. No menu suspenso Grupo de segurança, selecione os grupos de segurança da VPC que o AWS Security Agent deve usar:

> **Important**
>
> Garanta que seus grupos de segurança permitam conexões de saída para que o AWS Security Agent realize testes de penetração.

(Opcional) Configurar o CloudWatch registro

Configure CloudWatch para armazenar registros de seus testes de penetração. Essa seção é opcional e é reduzida por padrão.

1. Expanda a seção de CloudWatch registros.
2. No menu suspenso Grupos de registros, selecione grupos de registros existentes CloudWatch
3. Se não selecionar nenhum grupo de log, o AWS Security Agent criará um grupo de log nomeado `/aws/securityagent/<agent name>/<pentest id>` com as permissões apropriadas.

> **Note**
>
> Certifique-se de que sua função do IAM tenha permissões para gravar no grupo de CloudWatch registros selecionado.

(Opcional) Configurar segredos para credenciais de teste

Se seu aplicativo exigir credenciais de autenticação para testes, o AWS Security Agent poderá recuperá-las com segurança do AWS Secrets Manager. Essa seção é opcional e é reduzida por padrão.

1. Expanda a seção Segredos.
2. Selecione os segredos do AWS Secrets Manager que contêm credenciais para seu aplicativo.
3. Ao configurar um teste de penetração no aplicativo web, você pode fazer referência a esses segredos para testes autenticados.

> **Important**
>
> As credenciais são criptografadas e armazenadas no AWS Secrets Manager. Certifique-se de que sua função do IAM tenha permissões para acessar o Secrets Manager for AWS Security Agent para usar essas credenciais durante o teste.

(Opcional) Configurar funções Lambda para credenciais de teste

Configure as funções do Lambda que possam fornecer credenciais para seu aplicativo durante o teste. Essa seção é opcional e é reduzida por padrão.

1. Expanda a seção de funções do Lambda.
2. Selecione as funções do Lambda que podem fornecer credenciais de autenticação para seu aplicativo.
3. O AWS Security Agent invocará essas funções durante os testes de penetração para obter credenciais dinamicamente.

> **Note**
>
> Certifique-se de que sua função do IAM tenha permissões para invocar as funções Lambda especificadas. As funções Lambda devem retornar credenciais no formato esperado para uso do AWS Security Agent durante o teste.

(Opcional) Configurar o bucket S3

Forneça detalhes do bucket do S3 se você planeja fazer upload de documentos ou artefatos para fornecer como entrada para o AWS Security Agent. Essa seção é opcional e é reduzida por padrão.

1. Expanda a seção de buckets do S3.
2. No campo Bucket, insira ou pesquise o nome do bucket do S3.

> **Note**
>
> Você também pode conectar GitHub repositórios posteriormente ou fazer upload de arquivos diretamente no aplicativo web. As informações fornecidas podem garantir uma cobertura completa, reduzir os falsos positivos e fornecer resultados acionáveis.

(Opcional) Configurar o acesso ao serviço

O AWS Security Agent exige uma função do IAM para acessar seus recursos da AWS (VPC, grupos de CloudWatch log, Secrets Manager, funções Lambda etc.) para testes de penetração. Essa seção é opcional e é reduzida por padrão.

1. Expanda a seção Acesso ao serviço.
2. Por padrão, o AWS Security Agent usa uma função padrão do IAM com as permissões necessárias para testes de penetração.
3. Para personalizar a função do IAM, selecione uma das seguintes opções:
   a. Criar função padrão — O AWS Security Agent cria automaticamente uma nova função do IAM com as permissões necessárias
   b. Usar uma função de serviço existente — Selecione uma função do IAM existente no menu suspenso
4. Se estiver usando uma função existente:

a. Escolha o menu suspenso em Escolher uma função existente

b. Selecione sua função do IAM na lista

c. Escolha o ícone de atualização para atualizar a lista, se necessário

> **Note**
>
> A função padrão do IAM inclui permissões para acessar recursos de VPC, grupos de CloudWatch registros, Secrets Manager, funções Lambda e outros serviços necessários para testes de penetração. É recomendável usar a função padrão do IAM, a menos que você tenha requisitos de segurança específicos.

Etapa 4: salvar e ativar o teste de penetração

Depois de definir todas as configurações necessárias, habilite o teste de penetração para seu agente do AWS Security Agent.

1. Revise todas as seções de configuração para garantir a precisão.
2. Escolha Salvar.
3. O AWS Security Agent valida sua configuração e cria os recursos necessários da AWS.

Próximas etapas

Depois de ativar o teste de penetração:

- Configure os escopos do teste de penetração no aplicativo web do AWS Security Agent
- Configurar preferências de notificação para resultados de testes
- Revise e responda aos resultados do teste de penetração à medida que forem descobertos
- (Opcional) Configurar repositórios adicionais para remediação de descobertas

Para obter mais informações sobre como executar e gerenciar testes de penetração, consulte a documentação do aplicativo web do AWS Security Agent.

Habilitar revisão de código de pull request para GitHub repositórios

A revisão de código do pull request agora está configurada como parte do assistente de configuração da revisão de código. Ao conectar GitHub repositórios ao seu espaço de agente, você pode habilitar comentários de pull request para repositórios individuais para que o AWS Security Agent analise automaticamente os pull requests e publique descobertas de segurança.

Para obter instruções completas de configuração, consultethe section called “Ativar revisão de código”.

Para obter informações sobre como as descobertas do pull request aparecem GitHub e como responder a elas, consultethe section called “Analise as descobertas de segurança do código em pull requests”.

Ativar revisão de código

Configure seu espaço do agente para permitir a revisão de código conectando o código-fonte de GitHub repositórios ou buckets do S3. A revisão de código analisa seu código-fonte em busca de vulnerabilidades de segurança e conformidade com os requisitos de segurança personalizados de sua organização.

Definir as configurações de revisão de código é uma Space-wide operação do Agente. As integrações e os buckets S3 que você conecta são compartilhados entre os recursos, incluindo análise de código e teste de penetração.

Depois de concluir a configuração, os usuários podem criar e executar análises de código no aplicativo web do AWS Security Agent para verificar os repositórios e as fontes do S3 em busca de problemas de segurança.

> **Note**
>
> Se você já tem GitHub repositórios conectados ao seu Espaço do Agente (por exemplo, por meio da configuração do teste de penetração), a revisão de código já está ativada. Você pode pular essa configuração e ir diretamente para o aplicativo da web para criar revisões de código. Consulte the section called “Crie uma revisão de código”.

Pré-requisitos

Antes de começar, verifique se você tem:

- Um espaço de agente criado no AWS Management Console (consultethe section called “Crie um espaço de agente”)
- Pelo menos uma das seguintes entradas de código-fonte:
  - Uma GitHub organização ou conta de usuário com o GitHub aplicativo AWS Security Agent instalado (consultethe section called “Conecte o AWS Security Agent aos GitHub repositórios”)
  - Um bucket do S3 contendo o código-fonte que você deseja revisar
- Permissões para configurar integrações para seu Espaço do Agente
- (Opcional) Pelo menos um requisito de segurança ativado em um pacote de requisitos de segurança se você planeja usar a validação de requisitos de segurança (consultethe section called “Gerencie os requisitos de segurança”)

Acesse o assistente de configuração da revisão de código

Navegue até a configuração de revisão de código do seu Espaço do Agente.

1. No console do AWS Security Agent, selecione seu espaço de agente.
2. Escolha Ativar revisão de código em um dos seguintes locais:
   - O cartão de revisão de código
   - Na guia Revisão de código e, em seguida, escolha Habilitar revisão de código

> **Tip**
>
> Se você quiser que o AWS Security Agent resolva o feedback de revisão de código automaticamente, habilite também a remediação de código. Para mais detalhes, consulte the section called “Permita que os usuários iniciem a remediação dos resultados do teste de penetração e da revisão de código”.

Você é direcionado para o assistente de configurações de revisão de código de instalação.

Etapa 1: conectar integrações, repositórios e buckets

Na primeira etapa do assistente, conecte as entradas do código-fonte e defina as configurações de revisão de código. Você deve adicionar pelo menos um GitHub repositório ou bucket do S3 para continuar.

> **Important**
>
> As integrações e os buckets S3 configurados aqui são compartilhados em seu Espaço do Agente. As mudanças se aplicam aos recursos de revisão de código e teste de penetração.

Connect GitHub repositórios

Adicione GitHub repositórios de suas GitHub organizações autorizadas ou contas de usuário. Escolher Adicionar abre o GitHub assistente Connect em duas etapas, onde você primeiro seleciona os repositórios e depois configura quais ações o AWS Security Agent pode realizar em cada um.

1. Na seção Integrações conectadas, escolha Adicionar.
2. Selecione o GitHub registro que contém os repositórios que você deseja revisar.
3. Na etapa Conectar GitHub repositórios, marque a caixa de seleção de cada repositório que você deseja conectar.
4. Escolha Avançar para acessar Gerenciar recursos.

> **Note**
>
> Os repositórios conectados são acessados somente para leitura para análise de código durante a revisão do código e o teste de penetração. Você configura ações de gravação, como publicar comentários de avaliação e abrir pull requests de remediação na próxima etapa.

> **Note**
>
> Se você ainda não registrou uma GitHub integração, escolha Configurações para navegar até a página de integrações, onde você pode autorizar o aplicativo AWS Security Agent GitHub . Para obter mais informações, consulte the section called “Conecte o AWS Security Agent aos GitHub repositórios”.

Configurar os GitHub recursos do repositório

Na etapa Gerenciar capacidades do GitHub assistente Connect, escolha o que o AWS Security Agent pode fazer em cada repositório. Os comentários de revisão de código e a correção de código são definidos de forma independente por repositório.

1. Para cada repositório, ative os comentários de revisão de código para que o AWS Security Agent publique descobertas de segurança como comentários em pull requests.
2. Para cada repositório, ative a correção de código para permitir que o AWS Security Agent corrija as descobertas. Quando ativados, os usuários do aplicativo web podem solicitar pull requests que corrigem as descobertas, e os autores do pull request podem comentar `@AWS-Security-Agent fix all findings.` para que o agente resolva seus comentários de revisão.
3. Escolha Salvar para aplicar suas seleções e retornar ao assistente de configuração da revisão de código.

Quando os comentários de revisão de código estão habilitados para um repositório:

- O AWS Security Agent analisa automaticamente as pull requests quando elas são marcadas como “Prontas para análise”. Os rascunhos de pull requests não são analisados.
- As descobertas de segurança são publicadas como comentários de revisão diretamente na pull request com orientações específicas de remediação.
- A análise usa suas configurações de revisão de código definidas (vulnerabilidades de segurança, requisitos personalizados ou ambos).

> **Note**
>
> Os comentários do pull request só estão disponíveis para GitHub repositórios privados.

Quando a remediação de código é habilitada para um repositório, os usuários de aplicativos web podem iniciar a remediação tanto para análise de código quanto para resultados de testes de penetração nesse repositório, e o AWS Security Agent entrega cada correção como uma pull request. Para obter mais informações, consulte the section called “Permita que os usuários iniciem a remediação dos resultados do teste de penetração e da revisão de código”.

Para obter mais informações sobre como as descobertas do pull request aparecem GitHub e como responder a elas, consultethe section called “Analise as descobertas de segurança do código em pull requests”.

Adicionar buckets S3

Adicione buckets do S3 contendo código-fonte ou recursos contextuais para análise de código.

1. Na seção S3 buckets, escolha Adicionar recurso S3.
2. Insira o URI do S3 para o bucket ou prefixo que contém seu código-fonte.
3. Escolha Adicionar.

> **Note**
>
> Você pode adicionar até 10 recursos do S3. Os buckets do S3 são compartilhados entre os recursos, incluindo análise de código e teste de penetração.

> **Tip**
>
> Você pode adicionar buckets do S3 que contêm código-fonte, arquivos de configuração, modelos de infraestrutura como código ou outros artefatos que você deseja que o AWS Security Agent analise em busca de problemas de segurança.

Definir configurações de revisão de código

Configure os tipos de problemas de segurança que o AWS Security Agent analisa durante as análises de código. Essa configuração se aplica a todos os repositórios e fontes com a revisão de código ativada neste Espaço do Agente.

1. Na seção Configurações de revisão de código, selecione uma das seguintes opções:
   - Validação de requisitos de segurança — Valide se o código está em conformidade com os requisitos de segurança que você habilitou em seus pacotes de requisitos de segurança.
   - Descobertas de vulnerabilidades de segurança — Identifique vulnerabilidades de segurança comuns no código.

- Requisitos de segurança e descobertas de vulnerabilidades — Analise o código para verificar a conformidade com os requisitos de segurança que você habilitou e as vulnerabilidades de segurança comuns. Essa é a configuração padrão.

> **Note**
>
> Quando a validação de requisitos de segurança é ativada, o AWS Security Agent verifica o código em relação aos requisitos de segurança que você habilitou em seus pacotes de requisitos de segurança. Se você selecionar a validação de requisitos de segurança, mas não tiver pelo menos um requisito de segurança habilitado, o AWS Security Agent não identificará descobertas baseadas em requisitos. Para obter mais informações sobre requisitos de segurança, consulte the section called “Gerencie os requisitos de segurança”.

1. Escolha Avançar para prosseguir com as configurações opcionais.

Etapa 2: configurações opcionais

Na segunda etapa do assistente, defina as configurações opcionais de CloudWatch registro e acesso ao serviço para seu ambiente de revisão de código.

(Opcional) Configurar CloudWatch registros

Configure grupos de CloudWatch registros para capturar e analisar o comportamento do aplicativo durante a revisão do código.

1. Expanda a seção de CloudWatch registros.
2. No menu suspenso Grupos de registros, selecione um ou mais grupos de CloudWatch registros existentes na sua conta da AWS.

> **Note**
>
> Se você não selecionar um grupo de registros, o AWS Security Agent cria automaticamente um grupo de registros padrão para armazenar registros de execução de revisão de código.

(Opcional) Configurar o acesso ao serviço

Configure a função de serviço do IAM que o AWS Security Agent usa para acessar seus recursos da AWS, como buckets e CloudWatch registros do S3 para análise de código.

1. Expanda a seção Acesso ao serviço.
2. Selecione uma das opções a seguir:
   - Crie uma função padrão — O AWS Security Agent cria automaticamente uma nova função do IAM com as permissões necessárias para a revisão do código.
   - Use uma função de serviço existente — Selecione uma função do IAM existente no menu suspenso.
3. Se estiver usando a função padrão, insira um nome de função de serviço. O nome deve ser exclusivo em todas as funções na conta, usar caracteres alfanuméricos e `+=,.@-_` caracteres e não pode incluir espaços.

> **Note**
>
> O nome da função de serviço tem um tamanho máximo de 64 caracteres. Uma função de serviço é criada automaticamente se você não selecionar uma função existente.

1. Escolha Salvar para concluir a configuração.

Após a configuração

Depois de concluir o assistente de configuração da revisão de código:

- O cartão de revisão de código na sua página do Agent Space mostra o status Pronto.
- Os usuários podem iniciar o aplicativo web e criar análises de código para verificar repositórios conectados e fontes do S3.
- Você pode modificar sua configuração a qualquer momento escolhendo Editar configuração na guia Revisão de código.

Editar configuração de revisão de código

Para modificar sua configuração de revisão de código após a configuração inicial:

1. Navegue até seu espaço de agente no console do AWS Security Agent.
2. Selecione a guia Revisão de código.
3. Escolha Editar configuração.
4. Atualize suas integrações, buckets S3, configurações de revisão de código, CloudWatch registros ou acesso ao serviço conforme necessário.
5. Escolha Salvar.

Próximas etapas

Depois de definir as configurações de revisão de código:

- Inicie o aplicativo web para criar e executar revisões de código (consultethe section called “Crie uma revisão de código”)
- Conecte GitHub repositórios adicionais ou buckets do S3 à medida que sua base de código cresce
- Configure pacotes de requisitos de segurança para validação específica da organização (consulte) the section called “Gerencie os requisitos de segurança”
- Analise como os resultados do pull request aparecem em GitHub (consultethe section called “Analise as descobertas de segurança do código em pull requests”)

Ative a modelagem de ameaças

Configure seu Agent Space para permitir a modelagem de ameaças conectando repositórios de código-fonte e configurando recursos da AWS. A modelagem de ameaças analisa a arquitetura do seu aplicativo e identifica ameaças à segurança a partir do código-fonte, dos documentos de design ou de ambos.

Definir configurações de modelagem de ameaças é uma Space-wide operação do Agente. As integrações e os buckets S3 que você conecta são compartilhados entre os recursos, incluindo modelagem de ameaças, revisão de código e teste de penetração.

Depois de concluir a configuração, os usuários podem criar e executar modelos de ameaças no aplicativo web do AWS Security Agent.

> **Note**
>
> Se você já tem repositórios ou buckets S3 conectados ao seu Agent Space (por exemplo, por meio de revisão de código ou configuração de teste de penetração), a modelagem de

ameaças pode já estar habilitada. Você pode acessar diretamente o aplicativo da web para criar um modelo de ameaça. Consulte the section called “Crie um modelo de ameaça”.

Pré-requisitos

Antes de começar, verifique se você tem:

- Um espaço de agente criado no AWS Management Console (consultethe section called “Crie um espaço de agente”)
- Permissões para configurar integrações para seu Espaço do Agente
- (Opcional) Uma organização GitHub GitLab, ou Bitbucket com o aplicativo AWS Security Agent instalado (consulte Conectar o AWS Security Agent aos GitHub repositórios, Conectar o AWS Security Agent aos GitLab repositórios ou Conectar o AWS Security Agent aos repositórios do Bitbucket)

Acesse o assistente de configuração de modelagem de ameaças

Navegue até a configuração de modelagem de ameaças do seu Agent Space.

1. No console do AWS Security Agent, selecione seu espaço de agente.
2. Escolha Configurar modelo de ameaça no cartão Modelo de ameaça ou na guia Modelo de ameaça.

Você é direcionado ao assistente Configurar modelo de ameaça, que tem duas etapas opcionais.

Etapa 1: Conectar repositórios de código-fonte (opcional)

Conecte os repositórios GitHub GitLab,, ou do Bitbucket para os quais você deseja habilitar a modelagem de ameaças. Os próprios modelos de ameaças são criados e visualizados no aplicativo web.

Important

As integrações configuradas aqui são compartilhadas em seu Espaço do Agente. As mudanças se aplicam aos recursos de modelagem de ameaças, revisão de código e teste de penetração.

Connect repositórios

1. Na seção Integrações conectadas, escolha Adicionar.
2. Selecione o registro que contém os repositórios que você deseja usar.
3. Marque a caixa de seleção para cada repositório que você deseja conectar.
4. Escolha Salvar para aplicar suas seleções.

> **Note**
>
> Se você ainda não registrou uma integração, escolha Configurações para navegar até a página de integrações, onde você pode autorizar o aplicativo AWS Security Agent. Para obter mais informações, consulte Conectar o AWS Security Agent aos GitHub repositórios, Conectar o AWS Security Agent aos GitLab repositórios ou Conectar o AWS Security Agent aos repositórios do Bitbucket.

1. Escolha Avançar para prosseguir com as configurações opcionais.

Etapa 2: configurações opcionais

Defina buckets, CloudWatch registros e configurações de acesso ao serviço do S3 para seu ambiente de modelagem de ameaças. Essas configurações são compartilhadas com outros recursos no seu Espaço do Agente.

Caçambas S3 (opcionais)

Adicione buckets do S3 contendo o código-fonte que você deseja que o agente use como contexto durante a modelagem de ameaças.

1. Na seção S3 buckets, escolha Adicionar recurso S3.
2. Insira o URI do S3 para o bucket ou prefixo.
3. Escolha Adicionar.

> **Note**
>
> Você pode adicionar até 10 recursos do S3.

CloudWatch registros (opcional)

Configure grupos de CloudWatch registros para capturar e analisar o comportamento do aplicativo durante a execução do modelo de ameaças.

1. Na seção de CloudWatch registros, selecione um ou mais grupos de CloudWatch registros existentes na sua conta da AWS.

Acesso ao serviço

Configure a função de serviço do IAM que o AWS Security Agent usa para acessar seus recursos da AWS, como buckets e CloudWatch registros do S3 para modelagem de ameaças. É necessária uma função de serviço para permitir a modelagem de ameaças.

1. Na seção Acesso ao serviço, selecione uma função de serviço do IAM existente no menu suspenso ou deixe-a em branco para que uma função de serviço seja criada automaticamente.

> **Note**
>
> Uma função de serviço será criada automaticamente se você não selecionar uma função existente.

1. Escolha Salvar para concluir a configuração.

Após a configuração

Depois de concluir a configuração da modelagem de ameaças:

- O cartão do modelo Threat na sua página do Agent Space mostra o status Pronto.
- Os usuários podem iniciar o aplicativo web e criar modelos de ameaças a partir do código-fonte conectado, dos documentos de escopo enviados, das páginas do Confluence ou de uma combinação dessas entradas.

Próximas etapas

Depois de ativar a modelagem de ameaças:

- Inicie o aplicativo web para criar e executar modelos de ameaças (consulte Criar um modelo de ameaça)
- Conecte repositórios adicionais ou buckets do S3 à medida que sua base de código cresce
- Conecte o Confluence para permitir a seleção de páginas como documentos de escopo (consulte Conectar o AWS Security Agent ao Confluence)

Permita que os usuários iniciem a remediação dos resultados do teste de penetração e da revisão de código

No AWS Management Console, você pode ativar a remediação automática para que os usuários do aplicativo web AWS Security Agent possam solicitar correções para uma descoberta específica. O AWS Security Agent entrega cada correção como uma GitHub pull request no repositório afetado.

Você ativa a remediação por repositório a partir de um Espaço do Agente. As guias Teste de penetração e Revisão de código abrem o mesmo assistente de configuração do repositório, então você pode usar qualquer uma das guias para gerenciar a configuração de correção automática ativada. A correção se aplica às descobertas de ambos os recursos, portanto, você só precisa habilitá-la uma vez por repositório.

Pré-requisitos

Antes de começar, verifique se você tem:

1. Teste de penetração ativado (consultethe section called “Ativar teste de penetração”) ou revisão de código (consultethe section called “Ativar revisão de código”)
2. Instalou e autorizou o GitHub aplicativo AWS Security Agent para sua GitHub organização (consultethe section called “Conecte o AWS Security Agent aos GitHub repositórios”)

Abra o assistente de configuração do repositório

Escolha o ponto de entrada que corresponde à forma como seus repositórios estão configurados.

Na guia Teste de penetração

Use esse caminho para adicionar GitHub repositórios para testes de penetração e configurar a remediação na mesma etapa.

1. Navegue até a página de visão geral do Agent Space.

2. Escolha a guia Teste de penetração.
3. Selecione um GitHub registro que seja proprietário de seus repositórios:
   a. Se você não associou nenhum GitHub registro ao Espaço do agente, escolha Adicionar na caixa de informações do Connect GitHub to AWS Security Agent para selecionar um registro.
   b. Se um ou mais GitHub registros já estiverem associados, escolha Adicionar na seção Integrações conectadas para selecionar outro.
4. Escolha Avançar para escolher GitHub repositórios.
5. Escolha Avançar para configurar os recursos do repositório.

Na guia Revisão de código

Use esse caminho quando seus repositórios já estiverem conectados para revisão de código.

1. Navegue até a página de visão geral do Agent Space.
2. Escolha a guia Revisão de código.
3. Na tabela de GitHub repositórios, escolha Editar para abrir o assistente de configuração do repositório.

Ative a correção automática e economize

Depois de acessar os recursos do repositório, siga um dos caminhos acima:

1. Na coluna Remediação automática ativada, marque cada repositório que você deseja remediar como Ativado.
2. Escolha Connect para salvar a configuração.

Agora, os usuários do aplicativo web podem começar a remediar as descobertas nesses repositórios, e o AWS Security Agent abrirá pull requests com as correções propostas.

Forneça recursos de agente a partir de um bucket S3

Você pode conceder ao AWS Security Agent acesso aos recursos em um bucket do S3, como documentos de API, documentos de modelos de ameaças e código-fonte que suportam testes de penetração.

Você concede ao AWS Security Agent acesso de leitura a um bucket do S3 no Console de Gerenciamento da AWS. No aplicativo web do Security Agent, os usuários selecionam recursos individuais do S3 conforme relevante.

1. Navegue até a página de visão geral do Agent Space
2. Selecione Ações e, em seguida, Editar configuração do teste de penetração
3. Na seção S3 buckets, defina o URI do S3 que contém o S3 Bucket. Você pode adicionar vários buckets do S3.

Habilite um domínio de aplicativo para testes de penetração

Antes de executar um teste de penetração em um aplicativo, você precisa adicionar um domínio de destino e verificar a propriedade. O AWS Security Agent só realizará testes de penetração em domínios verificados.

> **Note**
>
> Você não precisa validar domínios auxiliares que seu aplicativo possa usar. Você só precisa validar o domínio no qual você executará ativamente os testes de penetração.

Etapa 1: adicionar um domínio de destino

Para adicionar um domínio de destino, navegue até a guia Teste de penetração na página de visão geral do Agent Space. Dependendo se você já configurou o teste de penetração, use um dos seguintes métodos:

- First-time configuração: Escolha Configurar teste de penetração para abrir o assistente de teste de penetração. Na primeira etapa, insira o domínio e escolha um método de verificação.
- Adicionar um domínio a uma configuração existente: na tabela Domínios de destino, escolha Adicionar domínio. Insira o domínio e escolha um método de verificação no modal.

Você pode adicionar um domínio base ou um subdomínio, como `example.com` ou`billing.example.com`. A AWS sugere o uso de um subdomínio em que você tenha permissão para criar `TXT` registros.

> **Note**
>
> Quando você verifica um domínio usando a verificação de rota DNS TXT ou HTTP, os subdomínios desse domínio são automaticamente cobertos. Por exemplo, a verificação `example.com` permite que você teste `api.example.com` ou `billing.example.com` sem verificação adicional. Para a verificação de VPC privada, o nome de domínio do endpoint de destino deve corresponder ao nome de domínio completo configurado para verificação.

Escolha um dos seguintes métodos de verificação:

- Registro DNS TXT: prove a propriedade do domínio criando um registro DNS TXT com seu provedor de DNS.
- Rota HTTP: prove a propriedade do domínio criando uma rota em seu servidor web que contém um token exclusivo fornecido pelo AWS Security Agent.
- VPC privada: só pode ser usada para testes de penetração de VPC privada. Verifica se o domínio é resolvido para um IP em um intervalo CIDR privado. O nome de domínio configurado deve corresponder ao nome de domínio completo de todos os endpoints de destino associados

Etapa 2: verificar a propriedade do domínio

Depois de adicionar o domínio, verifique a propriedade usando o método selecionado. Você pode acionar a verificação a qualquer momento na tabela de domínios de destino selecionando o domínio e escolhendo Verificar.

Verifique usando um registro DNS TXT

O AWS Security Agent gera um valor de registro DNS TXT. Você deve adicionar esse registro ao seu provedor de DNS para provar a propriedade.

Se o domínio estiver registrado no Route 53 (mesma conta da AWS):

O AWS Security Agent pode criar o registro DNS automaticamente. Selecione o domínio na tabela Domínios de destino e escolha a One-click verificação. O AWS Security Agent cria o registro de validação do DNS e conclui a verificação automaticamente.

Se o domínio estiver registrado com outro provedor de DNS:

1. Copie o valor do registro TXT fornecido pelo AWS Security Agent.

2. Adicione o registro TXT com seu registrador de DNS.
3. Retorne à tabela Domínios de destino, selecione o domínio e escolha Verificar.

Verifique usando a validação da rota HTTP

Esse método prova a propriedade do domínio colocando um token exclusivo em seu servidor web. Somente proprietários de domínios ou administradores autorizados da Web podem criar rotas em um servidor da Web, o que prova a propriedade.

1. Crie um arquivo no seguinte caminho no seu servidor web:

```
.well-known/aws/securityagent-domain-verification.json
```

2. Coloque o token fornecido pelo AWS Security Agent no arquivo usando este formato:

```
{
  "tokens": ["<insert-token>"]
}
```

3. Retorne à tabela Domínios de destino, selecione o domínio e escolha Verificar. O AWS Security Agent envia uma solicitação HTTPS GET para a URL de verificação e valida o token.
4. Se o domínio estiver acessível na Internet pública, certifique-se de que seu domínio tenha um certificado SSL válido antes de executar a verificação.

> **Note**
>
> Se seu domínio estiver registrado em vários espaços de agente e você estiver usando a validação de rota HTTP, você poderá colocar os tokens dos dois espaços de agente na mesma `tokens` matriz.

Ignorar a verificação de propriedade do domínio

Clientes que têm autorização para realizar testes de penetração em um endpoint, mas não conseguem concluir a verificação de propriedade, podem solicitar nossa verificação manual. Abra um caso de suporte ao cliente para fazer essa solicitação. Sua solicitação deve incluir seu caso de uso comercial e sua justificativa para a verificação manual da propriedade (ou seja, por que você está autorizado a realizar testes de penetração no endpoint).

Domínios-alvo gerenciados usados para testes de penetração

No AWS Management Console, você pode adicionar e gerenciar domínios de destino consumidos por espaços de agentes para testes de penetração. Esses domínios-alvo precisarão ser verificados antes de serem usados em um teste de penetração. Para obter mais informações sobre a verificação de domínios de destino, consulte the section called “Ativar teste de penetração”

Pré-requisitos

Antes de começar, verifique se você tem:

1. Teste de penetração ativado (consultethe section called “Ativar teste de penetração”)

Gerencie recursos do domínio de destino

- Navegue até a página de visão geral dos domínios de destino.
- Você deve ver todos os recursos do domínio de destino associados à sua conta
  - Se você não encontrar nenhum domínio de destino associado à sua conta, siga as etapas the section called “Ativar teste de penetração” para criar e verificar um domínio de destino
- Os domínios de destino podem ser reutilizados entre os espaços do agente e compartilhar o status de verificação
  - Para adicionar um domínio de destino existente a um espaço do agente, navegue até a guia Teste de penetração do espaço do agente. Selecione Adicionar domínio e escolha o domínio desejado em Selecionar entre os domínios registrados anteriormente disponíveis no campo do nome do domínio
  - Os domínios de destino devem ser associados a um espaço de agente antes de poderem ser usados em um teste de penetração
- Remover um domínio de um espaço de agente não exclui o domínio. O domínio de destino associado pode ser excluído permanentemente da página de visão geral dos Domínios de Destino

Verifique os domínios de destino

Para que um domínio alvo seja usado em um teste de penetração, ele deve primeiro ser verificado usando um dos métodos abaixo:

- Domínios do Route 53 (mesma conta da AWS): escolha a One-click verificação. O AWS Security Agent cria automaticamente o registro DNS e conclui a verificação.

- DNS TXT (outros provedores de DNS): copie o token de verificação, adicione o registro TXT ao seu registrador de DNS, selecione o domínio e escolha Verificar.
- Rota HTTP: coloque o token de verificação no caminho de rota necessário em seu servidor web, selecione o domínio e escolha Verificar. Para obter detalhes, consulte the section called “Habilite um domínio de aplicativo para testes de penetração”.
- VPC privada: verifica se o IP do domínio de destino está dentro de um intervalo CIDR privado (consulte the section called “Conecte o agente aos recursos privados da VPC” para obter uma lista de intervalos CIDR privados). O nome de domínio do endpoint alvo do teste de penetração deve corresponder ao nome de domínio completo configurado para verificação. Só pode ser usado para testes de penetração de VPC privados

> **Note**
>
> Para verificação de DNS TXT, HTTP route e Route 53, os subdomínios de um domínio verificado são automaticamente cobertos e não exigem verificação separada. Para a verificação de VPC privada, cada domínio deve ser verificado individualmente.

Gerencie os requisitos de segurança

Defina os requisitos de segurança que o AWS Security Agent avalia durante as análises de design e código, usando AWS-managed pacotes, pacotes personalizados ou requisitos importados de sua própria documentação.

Gerencie os requisitos de segurança

Organize os requisitos de segurança que o AWS Security Agent usa para analisar seus aplicativos em pacotes de requisitos de segurança. Um pacote é uma coleção de requisitos de segurança. Você pode AWS habilitar pacotes gerenciados, criar seus próprios pacotes personalizados e gerar requisitos para um pacote personalizado fazendo o upload de sua documentação de segurança.

Visão geral do

Um requisito de segurança define um padrão ou política de segurança com o qual o AWS Security Agent avalia seu aplicativo durante análises de design e análises de código. Quando um design ou código não atende a um requisito, o AWS Security Agent relata uma descoberta. Cada requisito

de segurança pertence a um pacote de requisitos de segurança. Os requisitos de segurança são compartilhados em todos os Agent Spaces e avaliados durante as análises de design e código.

AWS O Security Agent fornece dois tipos de pacotes, mostrados em guias separadas:

- Pacotes gerenciados de requisitos de segurança AWS— pacotes de requisitos de segurança fornecidos com base nos padrões e nas melhores práticas do setor. Você pode ativar esses pacotes instantaneamente para avaliar as políticas de segurança comuns sem configuração personalizada. Os requisitos em um pacote gerenciado são somente para leitura. Você pode usar um requisito AWS gerenciado como modelo para um requisito personalizado. Para ver a lista de pacotes gerenciados disponíveis, consulte pacotes de requisitos de segurança AWS gerenciados.
- Pacotes de requisitos de segurança personalizados — Pacotes que você cria para aplicar as políticas e padrões específicos da sua organização. Você cria os requisitos em um pacote personalizado do zero, personaliza os requisitos AWS gerenciados como ponto de partida ou os gera fazendo o upload da documentação de segurança.

> **Note**
>
> Os pacotes de estrutura de conformidade foram projetados para ajudar a identificar os requisitos de segurança que podem ser relevantes para a conformidade com determinadas estruturas. Eles não avaliam sua conformidade nem garantem que você passará por uma auditoria.

Você ativa e desativa pacotes como uma unidade. Quando um pacote é ativado, o AWS Security Agent avalia seus aplicativos em relação aos requisitos desse pacote durante as análises de design e código.

> **Note**
>
> A ativação ou desativação de um pacote se aplica a novas revisões de design e revisões de código. As avaliações existentes não são afetadas.

Ativar ou desativar um pacote gerenciado

Ative um pacote AWS gerenciado para avaliar seus aplicativos em relação a um conjunto de requisitos AWS de segurança mantidos. Desative-o quando não precisar mais dele.

1. No AWS console, navegue até AWS Security Agent.
2. No painel de navegação, escolha Requisitos de segurança.
3. Escolha a guia Pacotes de requisitos de segurança gerenciados.
4. Selecione o pacote que você deseja ativar ou desativar.
5. Execute um destes procedimentos:
   a. Para ativar o pacote, escolha Ativar pacote.
   b. Para desativar o pacote, escolha Desativar pacote.

> **Tip**
>
> Escolha um pacote para ver os requisitos de segurança que ele contém. Escolha um requisito para ver sua definição completa, incluindo sua aplicabilidade, critérios de conformidade e orientação de remediação.

Crie um pacote personalizado

Crie um pacote personalizado para agrupar os requisitos de segurança específicos da sua organização. Depois de criar o pacote, adicione requisitos a ele manualmente, personalize um requisito AWS gerenciado ou faça upload de documentos para gerar requisitos.

1. No AWS console, navegue até AWS Security Agent.
2. No painel de navegação, escolha Requisitos de segurança.
3. Escolha a guia Pacotes de requisitos de segurança personalizados.
4. Escolha Criar pacote de requisitos de segurança.
5. Insira o nome do pacote de requisitos de segurança e uma descrição opcional.
6. Escolha Criar pacote de requisitos de segurança.

> **Note**
>
> Depois de criar um pacote, você pode adicionar ou carregar documentos de origem para gerar os requisitos de segurança para seu pacote.

Adicione um requisito de segurança manualmente

Adicione um requisito de segurança a um pacote personalizado para definir um padrão de segurança que o AWS Security Agent aplica.

1. No AWS console, navegue até AWS Security Agent.
2. No painel de navegação, escolha Requisitos de segurança.
3. Escolha a guia Pacotes de requisitos de segurança personalizados e, em seguida, escolha o pacote ao qual você deseja adicionar um requisito.
4. Escolha Adicionar requisito manualmente.
5. Configure os campos a seguir.
   a. Nome do requisito de segurança — insira um nome descritivo que identifique o controle de segurança, por exemplo `enforce-encryption-at-rest` (máximo de 80 caracteres).
   b. Descrição — Forneça uma breve descrição do que esse requisito de segurança verifica (máximo de 500 caracteres).
   c. Aplicabilidade — Descreva os cenários, tipos de sistema ou condições em que esse requisito de segurança deve ser avaliado. Inclua os cenários em que o requisito deve ser marcado como NOT_APPLICABLE (máximo de 10.000 caracteres).
   d. Critérios de conformidade — defina o que constitui conformidade versus não conformidade para esse requisito de segurança. Forneça os indicadores, exemplos e detalhes técnicos que o AWS Security Agent procura ao avaliar a conformidade (máximo de 10.000 caracteres).
   e. Orientação de remediação (opcional) — forneça orientação sobre como corrigir violações, incluindo links para a documentação ou padrões internos da sua organização (máximo de 10.000 caracteres).
6. Execute um destes procedimentos:
   a. Para criar o requisito sem ativá-lo, escolha Criar requisito de segurança.
   b. Para criar o requisito e ativá-lo, escolha Criar e ativar o requisito de segurança.

> **Note**
>
> Um requisito é avaliado durante as análises de segurança somente quando o pacote que o contém está ativado. Para controlar se um pacote é avaliado, ative ou desative o pacote.

Personalize um AWS-requisito de segurança gerenciado

Crie um requisito de segurança personalizado que use um requisito AWS gerenciado como ponto de partida. AWS O Security Agent preenche previamente os detalhes do requisito gerenciado e você os edita para adequá-los à sua organização.

1. No AWS console, navegue até AWS Security Agent.
2. No painel de navegação, escolha Requisitos de segurança.
3. Escolha a guia Pacotes de requisitos de segurança personalizados e, em seguida, escolha o pacote personalizado ao qual adicionar o requisito.
4. Escolha Criar requisito de segurança personalizado.
5. Para preencher previamente o formulário, na seção Personalizar um requisito AWS de segurança gerenciado, pesquise e selecione um requisito AWS gerenciado para usar como modelo.
6. Edite qualquer um dos campos para atender às necessidades da sua organização.
7. Execute um destes procedimentos:
   a. Para criar o requisito sem ativá-lo, escolha Criar requisito de segurança.
   b. Para criar e ativar imediatamente o requisito, escolha Criar e ativar o requisito de segurança.

> **Note**
>
> A personalização AWS de um requisito gerenciado cria um requisito de segurança personalizado independente. Alterações AWS posteriores no requisito gerenciado não afetam sua versão personalizada.

Gere requisitos fazendo o upload de documentos

Gere os requisitos de segurança para um pacote personalizado a partir de sua documentação de segurança existente, em vez de escrever cada requisito manualmente. AWS O Security Agent lê os documentos que você carrega, identifica o conteúdo relevante para a segurança e gera requisitos estruturados que você pode revisar e editar. Para ver o procedimento completo, consulte Gerar requisitos de segurança a partir de documentos. Para obter orientação sobre o que incluir nos documentos que você carrega, consulte Preparar documentos para geração de requisitos.

> **Important**
>
> O upload de novos documentos de origem regenera todos os requisitos do pacote. Todos os requisitos existentes, incluindo aqueles que você adicionou manualmente, são substituídos.

Editar um requisito de segurança personalizado

Modifique um requisito de segurança personalizado para atualizar sua definição, critérios ou diretrizes de remediação. Você não pode editar os requisitos em um pacote AWS gerenciado.

1. No AWS console, navegue até AWS Security Agent.
2. No painel de navegação, escolha Requisitos de segurança.
3. Escolha a guia Pacotes de requisitos de segurança personalizados e, em seguida, escolha o pacote que contém o requisito.
4. Selecione o requisito que você deseja editar e escolha Editar requisito de segurança personalizado.
5. Atualize os campos de requisitos conforme necessário. O nome do requisito de segurança não pode ser alterado após a criação.
6. Escolha Atualizar requisito de segurança.

> **Note**
>
> As alterações em um requisito de segurança se aplicam a novas revisões de design e revisões de código. As avaliações existentes não são afetadas.

Ativar ou desativar um pacote

Ative ou desative um pacote de requisitos de segurança para controlar se o AWS Security Agent avalia os requisitos que ele contém. Você ativa e desativa pacotes como uma unidade.

1. No AWS console, navegue até AWS Security Agent.
2. No painel de navegação, escolha Requisitos de segurança.
3. Escolha a guia para o tipo de pacote e, em seguida, selecione o pacote.

4. Execute um destes procedimentos:
   a. Para ativar o pacote, escolha Ativar pacote.
   b. Para desativar o pacote, escolha Desativar pacote.

> **Note**
>
> Quando um pacote é ativado, os requisitos do pacote são avaliados durante as análises de segurança. Quando um pacote é desativado, seus requisitos não são avaliados.

Excluir um requisito de segurança personalizado

Exclua um requisito de segurança personalizado quando não quiser mais que o AWS Security Agent o avalie.

1. No AWS console, navegue até AWS Security Agent.
2. No painel de navegação, escolha Requisitos de segurança.
3. Escolha a guia Pacotes de requisitos de segurança personalizados e, em seguida, escolha o pacote que contém o requisito.
4. Selecione um ou mais requisitos de segurança e escolha Excluir requisito de segurança.
5. Confirme a exclusão.

> **Note**
>
> Quando você exclui os requisitos de segurança, eles não são mais avaliados em futuras análises. Os requisitos permanecem visíveis em análises anteriores como resultados somente para leitura.

Melhores práticas para definir requisitos de segurança

Ao criar um requisito de segurança, siga estas diretrizes para ajudar o AWS Security Agent a avaliar seus aplicativos com precisão e fornecer descobertas práticas.

Nome do requisito de segurança — Use um nome claro e específico que identifique o controle de segurança. Evite termos genéricos que não transmitam a finalidade do requisito.

Descrição — Explique o que o controle verifica e o risco que ele mitiga. Isso ajuda os usuários a entender por que o requisito é importante.

Aplicabilidade — Descreva as cargas de trabalho, os tipos de sistema ou as condições em que o requisito deve ser avaliado. Indique quando o requisito deve ser marcado como NOT_APPLICABLE, com cenários específicos, para evitar falsos positivos. Frases como “Esse controle se aplica a TODAS as cargas de trabalho que...” e “Marcar como NOT_APPLICABLE se...” definem limites claros do escopo.

Critérios de conformidade — estruture isso em duas partes: o que demonstra conformidade e o que indica não conformidade. Seja específico com indicadores técnicos e inclua casos extremos. Comece com “Um design está em conformidade se demonstrar...” seguido por detalhes técnicos e, em seguida, “Um design não está em conformidade se...” com os padrões que indicam uma violação.

Orientação de remediação — forneça orientação com detalhes técnicos e exemplos de configuração. Inclua links para a documentação interna da sua organização para que os desenvolvedores tenham os recursos necessários para corrigir as violações.

Próximas etapas

Depois de configurar seus pacotes de requisitos de segurança:

- Crie uma revisão de design ou revisão de código para avaliar seu aplicativo em relação aos pacotes que você ativou.
- Ative o teste de penetração para complementar suas análises de design e código.
- Conceda aos usuários acesso ao aplicativo web do AWS Security Agent.

Pacotes de requisitos de segurança gerenciados pela AWS

Um pacote AWS gerenciado de requisitos de segurança é um conjunto de requisitos de segurança AWS criado e mantido com base nos padrões e nas melhores práticas do setor. Você permite que um pacote gerenciado avalie seus aplicativos em relação aos requisitos durante as análises de design e de código, sem precisar escrever os requisitos por conta própria.

Os requisitos de segurança em um pacote gerenciado são somente para leitura. Você não pode alterá-los. Você pode usar um requisito AWS gerenciado como modelo para criar um requisito personalizado e, em seguida, editar a cópia para adequá-la à sua organização. Para obter mais informações, consulte Gerenciar requisitos de segurança.

AWS atualiza os pacotes gerenciados ao longo do tempo. Quando AWS adiciona ou revisa um requisito em um pacote gerenciado, a alteração se aplica a todas as contas que têm o pacote ativado.

> **Note**
>
> Os pacotes de estrutura de conformidade foram projetados para ajudar a identificar os requisitos de segurança que podem ser relevantes para a conformidade com determinadas estruturas. Eles não avaliam sua conformidade nem garantem que você passará por uma auditoria.

Pacote gerenciado pela AWS: ASA Base Pack

Um conjunto básico de requisitos de segurança que se aplicam à maioria das cargas de trabalho. Este pacote aborda questões comuns de segurança de aplicativos, como autenticação e autorização, proteção de dados, registro e validação de entrada. Ative esse pacote para estabelecer uma linha de base de segurança para suas análises de design e código.

Pacote gerenciado pela AWS: AWS Well-Architected Pack

Requisitos de segurança derivados do pilar de segurança da AWS Well-Architected Estrutura. Este pacote avalia seu aplicativo em relação às AWS diretrizes para proteção de dados, gerenciamento de identidades e permissões e detecção e resposta a eventos de segurança. Para obter mais informações sobre a estrutura, consulte Security Pillar — AWS Well-Architected Framework.

Pacote gerenciado pela AWS: NIST CSF Pack

Requisitos de segurança derivados do NIST Cybersecurity Framework (CSF). Este pacote avalia seu aplicativo em relação às áreas de controle extraídas da estrutura, como gerenciamento de identidade e acesso, segurança de dados e tecnologia de proteção. Habilite este pacote para alinhar suas análises de design e código com o NIST CSF.

Pacote gerenciado pela AWS: Pacote PCI DSS

Requisitos de segurança derivados do Padrão de Segurança de Dados do Setor de Cartões de Pagamento (PCI DSS). Esse pacote avalia seu aplicativo em relação às áreas de controle relevantes para lidar com dados de titulares de cartões, como controle de acesso, criptografia de dados em

trânsito e em repouso e registro. Ative esse pacote para alinhar suas revisões de design e código com o PCI DSS.

Gere requisitos de segurança a partir de documentos

Gere os requisitos de segurança para um pacote personalizado fazendo o upload da documentação de segurança existente. AWS O Security Agent lê os documentos que você carrega, identifica o conteúdo relevante para a segurança e gera requisitos de segurança estruturados com aplicabilidade, critérios de conformidade e diretrizes de remediação. Você pode revisar e editar os requisitos gerados antes de serem usados nas revisões.

Use isso em vez de escrever cada requisito manualmente quando você já tiver políticas, padrões ou diretrizes de segurança documentados. Para obter orientação sobre o que incluir nos documentos que você carrega, consulte Preparar documentos para geração de requisitos.

Formatos de arquivo compatíveis

Você pode fazer upload dos seguintes formatos de arquivo:

- DOC
- DOCX
- MD
- PDF
- TXT

Gere requisitos

1. No AWS console, navegue até AWS Security Agent.
2. No painel de navegação, escolha Requisitos de segurança.
3. Escolha a guia Pacotes de requisitos de segurança personalizados.
4. Crie um pacote ou escolha um pacote personalizado existente para gerar requisitos.
5. Escolha Escolher arquivos ou arraste e solte seus documentos na área de upload.
6. Escolha Gerar requisitos.
7. Aguarde até que o AWS Security Agent processe os documentos. A geração é executada em segundo plano e pode levar alguns minutos para arquivos grandes. Você não pode iniciar outro upload para o pacote enquanto a geração estiver em andamento.

8. Analise os requisitos de segurança gerados. Edite, exclua ou adicione requisitos conforme necessário. Ative o pacote quando quiser que o AWS Security Agent avalie seus requisitos durante as análises de segurança.

> **Note**
>
> AWS O Security Agent gera requisitos no nível da carga de trabalho e se concentra no conteúdo relevante para a segurança em seus documentos. O número de requisitos que ele gera depende do conteúdo que você fornece. Analise os requisitos gerados para confirmar que eles refletem sua intenção.

Substitua os requisitos por um novo upload

O upload de novos documentos de origem regenera os requisitos de um pacote.

> **Important**
>
> Quando você carrega novos documentos de origem em um pacote, o AWS Security Agent regenera todos os requisitos desse pacote. Todos os requisitos existentes, incluindo aqueles que você adicionou manualmente, são substituídos. Para manter seus requisitos atuais, não faça upload de novos documentos.

1. No AWS console, navegue até AWS Security Agent.
2. No painel de navegação, escolha Requisitos de segurança.
3. Escolha a guia Pacotes de requisitos de segurança personalizados e, em seguida, escolha o pacote.
4. Inicie um novo upload e reconheça que o upload de novos documentos de origem substitui os requisitos existentes.
5. Escolha seus arquivos e, em seguida, escolha Gerar requisitos.

Próximas etapas

- Revise e habilite os requisitos gerados. Consulte Gerenciar requisitos de segurança.
- Crie uma revisão de design ou revisão de código para avaliar seu aplicativo em relação ao pacote.

Preparar documentos para geração de requisitos

Quando você gera requisitos de segurança a partir de documentos, o AWS Security Agent lê seus documentos e produz requisitos estruturados a partir do conteúdo relevante de segurança encontrado. Você não precisa formatar seus documentos de nenhuma forma específica nem pré-escrever a aplicabilidade, os critérios de conformidade e as diretrizes de remediação. AWS O Security Agent os deriva do conteúdo que você fornece. Faça o upload da documentação de segurança que você já tem, escrita com suas próprias palavras.

Esta página descreve o que incluir para que o AWS Security Agent gere requisitos precisos e úteis. Para o procedimento de upload e os limites de arquivos, consulte Gerar requisitos de segurança a partir de documentos.

O que incluir

AWS O Security Agent procura conteúdo que descreva suas expectativas de segurança. Documentos que abrangem os tópicos a seguir produzem os requisitos mais rigorosos:

- Controle de acesso e autorização
- Autenticação
- Proteção e criptografia de dados
- Segurança de rede
- Registro e auditoria
- Resposta a incidentes
- Gerenciamento de vulnerabilidades e patches
- Obrigações de conformidade que se aplicam às suas cargas de trabalho

Os bons documentos de origem incluem políticas de segurança, padrões de engenharia, catálogos de controle, diretrizes de arquitetura e padrões de codificação segura.

Escreva no nível da carga de trabalho

AWS O Security Agent gera requisitos que se aplicam a uma carga de trabalho ou aplicativo individual, o mesmo escopo que ele avalia durante as análises de design e código. O conteúdo que descreve como criar e operar uma carga de trabalho segura gera requisitos. Organization-wide tópicos de governança, como estratégia de várias contas, gerenciamento de usuários raiz e

configuração de registro em nível de conta, estão fora desse escopo e não geram requisitos de carga de trabalho.

Descreva o resultado, não uma única implementação

Indique o resultado de segurança que você espera, em vez de um único produto ou configuração prescrita. AWS O Security Agent gera requisitos que se concentram na capacidade de segurança necessária e permitem mais de uma implementação válida. Por exemplo, “dados em repouso são criptografados” gera um requisito mais claro e mais amplamente aplicável do que uma declaração que nomeia um serviço ou configuração específica.

Seja específico sobre a aparência bonita

Quanto mais concretamente seus documentos descreverem o comportamento esperado, melhor o AWS Security Agent poderá definir os critérios de conformidade. Sempre que possível, descreva o que um design compatível demonstra e o que indica uma violação. Declarações vagas ou puramente ambiciosas geram menos e mais fracas exigências do que aquelas concretas e relevantes para a carga de trabalho.

Revise o que você recebe de volta

Cada requisito gerado inclui um nome, aplicabilidade, critérios de conformidade e orientação de remediação que o AWS Security Agent derivou de seus documentos. Analise os requisitos gerados, edite os que precisam ser refinados e habilite aqueles que você deseja que o AWS Security Agent avalie. Para obter mais informações, consulte Gerenciar requisitos de segurança.

Gerencie usuários e acessos

Controle quem pode acessar cada espaço do agente e configure as funções do IAM e as chaves de criptografia que o AWS Security Agent usa.

Conceda aos usuários acesso ao aplicativo web AWS Security Agent

O AWS Security Agent fornece dois métodos para os usuários acessarem o aplicativo web, dependendo de como você configurou seu espaço do agente durante a configuração.

Visão geral dos métodos de acesso

IAM Identity Center (SSO) — Se você ativou o IAM Identity Center ao criar seu Agent Space, os usuários poderão acessar o aplicativo web diretamente por meio do SSO. Você atribui usuários ao

Espaço do Agente por meio do console do AWS Security Agent, e os usuários fazem login usando suas credenciais do Identity Center.

Acesso de administrador — Usuários com acesso ao AWS Console podem iniciar o aplicativo web por meio de um link de acesso de administrador na página de visão geral do Agent Space no AWS Management Console.

Conceda acesso com o IAM Identity Center (SSO)

Se você configurou seu espaço do agente com o IAM Identity Center, você pode atribuir usuários ao espaço do agente usando o console do AWS Security Agent.

Atribua usuários por meio do console do AWS Security Agent

1. No AWS Security Agent Management Console, navegue até seu espaço de agente.
2. Selecione a guia Aplicativo Web.
3. Na tabela Usuários, escolha Adicionar usuários.
4. Selecione usuários existentes no IAM Identity Center ou crie novos usuários.
5. Confirme as atribuições do usuário.

> **Tip**
>
> Os usuários atribuídos ao Agent Space podem acessar o aplicativo web fazendo login por meio do IAM Identity Center com suas credenciais de SSO.

Acesse o aplicativo web com SSO

Depois que os usuários forem atribuídos ao Espaço do Agente:

1. Os usuários navegam até a URL do aplicativo web do Agent Space.

> **Tip**
>
> Encontre o URL do aplicativo web na página de detalhes do Agent Space no console do AWS Security Agent selecionando Copiar URL do aplicativo web. Os usuários devem marcar esse URL para facilitar o acesso.

2. Os usuários fazem login usando suas credenciais de SSO.
3. Após a autenticação, os usuários podem selecionar o Espaço do Agente e começar a realizar avaliações de segurança.

Conceda acesso com IAM-only acesso

Se você configurou seu Agent Space com IAM-only acesso, os usuários com acesso ao AWS Console podem iniciar o aplicativo web por meio de um link de acesso administrativo.

Use o link de acesso do administrador

1. Faça login no console do AWS Security Agent.
2. Navegue até o Espaço do Agente que você deseja acessar.
3. Na guia do aplicativo Web da página inicial do Agent Space, escolha o botão de acesso do administrador para iniciar o aplicativo web.
4. O aplicativo web é aberto em uma nova guia com o usuário autenticado automaticamente.

> **Note**
>
> O link de acesso do administrador só está disponível para usuários que já estão autenticados no Console da AWS com as permissões apropriadas do AWS Security Agent. Esse método não exige a configuração do IAM Identity Center.

Próximas etapas

Depois de conceder aos usuários acesso ao aplicativo web:

- Os usuários podem criar e gerenciar configurações e execuções de testes de penetração
- Os usuários podem criar e gerenciar revisões de design
- Os usuários podem ver as descobertas de segurança e as diretrizes de remediação
- Configurar preferências de notificação para descobertas de segurança

Crie uma função do IAM para o AWS Security Agent

O AWS Security Agent usa funções do IAM de três maneiras:

1. Função do aplicativo: usada ao criar o aplicativo AWS Security Agent. Para casos de uso do IAM Identity Center e do link de acesso administrativo, o serviço assume essa função de conceder aos WebApp usuários permissões para interagir com as APIs do AWS Security Agent.
2. Função do serviço de teste de penetração: especificada ao criar espaços de agente como uma lista de funções disponíveis. Posteriormente, WebApp os usuários selecionam uma dessas funções ao criar um teste de penetração. O serviço AWS Security Agent assume essa função para acessar seus recursos da AWS durante o teste.
3. Papel do ator: usado para autenticar e autorizar solicitações para seu aplicativo web de destino (por exemplo, APIs do AWS API Gateway). Essas funções são fornecidas durante a criação do espaço do agente. O agente do AWS Security Agent assume funções de ator para interagir com seu aplicativo de destino.

Função do aplicativo

A função do aplicativo é usada ao criar seu aplicativo do AWS Security Agent no serviço. Para cenários de autenticação do IAM Identity Center e do link de acesso do administrador, o serviço AWS Security Agent assume essa função para conceder aos WebApp usuários as permissões necessárias para interagir com as APIs do AWS Security Agent.

Permissões obrigatórias

Essa função precisa de permissões para:

- Invoque operações de API do AWS Security Agent
- Leia e grave dados de configuração do aplicativo
- Acesse as informações da sessão do usuário
- Gerenciar tokens de autenticação para WebApp usuários

Política de confiança

A política de confiança deve permitir que o serviço AWS Security Agent assuma essa função:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Política de permissões

A função deve incluir permissões para:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "securityagent:GetApplication",
        "securityagent:UpdateApplication",
        "securityagent:ListAgentInstances",
        "securityagent:CreatePentestSession"
      ],
      "Resource": "arn:aws:securityagent:*:*:application/*"
    }
  ]
}
```

> **Note**
>
> Personalize as permissões com base nos requisitos específicos do seu aplicativo e no princípio do menor privilégio.

Função do serviço de teste de penetração

A função do serviço de teste de penetração é especificada ao criar espaços de agente como uma lista de funções disponíveis. Quando WebApp os usuários criam um teste de penetração, eles selecionam uma dessas funções. O serviço AWS Security Agent então assume essa função para acessar e testar seus recursos da AWS.

Permissões obrigatórias

Essa função precisa de permissões para acessar e analisar seus recursos da AWS durante o teste de penetração:

- Leia e descreva as configurações de VPC e a topologia de rede
- Inspecione instâncias do EC2, grupos de segurança e ACLs de rede
- Analise as políticas do IAM e as permissões de recursos
- Leia CloudWatch registros e métricas
- Acesse as configurações de serviços da AWS relevantes para testes de segurança

Política de confiança

A política de confiança deve permitir que o serviço AWS Security Agent assuma essa função nas operações de teste de penetração:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "your-external-id"
        }
      }
    }
  ]
}
```

> **Note**
>
> Use uma ID externa para maior segurança ao permitir o acesso entre contas ou serviços.

Política de permissões

A função deve incluir acesso somente de leitura aos seus recursos da AWS. Considere usar estas políticas gerenciadas:

- `SecurityAudit`- Política gerenciada pela AWS para auditoria de segurança
- `ViewOnlyAccess`- Read-only acesso à maioria dos serviços da AWS

Ou crie uma política personalizada com permissões específicas:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:Describe*",
        "vpc:Describe*",
        "iam:Get*",
        "iam:List*",
        "logs:DescribeLogGroups",
        "logs:DescribeLogStreams",
        "cloudwatch:Describe*",
        "cloudwatch:Get*",
        "cloudwatch:List*",
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": "*"
    }
  ]
}
```

Important

Conceda somente as permissões mínimas necessárias para o escopo do seu teste de penetração. Revise e ajuste as permissões com base nos serviços da AWS que você deseja incluir nos testes de segurança.

Papel do ator

A função de ator é usada para autenticar e autorizar solicitações para seu aplicativo web de destino durante o teste de penetração. Essas funções são fornecidas durante a criação do espaço do agente, e o agente do AWS Security Agent presume que elas interajam com os endpoints do aplicativo de destino (como APIs do AWS API Gateway, URLs da função Lambda ou outros aplicativos). AWS-hosted

Permissões obrigatórias

Essa função precisa de permissões para:

- Invoque os endpoints do API Gateway
- Execute funções Lambda
- Acesse recursos da AWS específicos do aplicativo
- Autentique-se com os mecanismos de autenticação do seu aplicativo de destino
- Execute operações HTTP nos endpoints do seu aplicativo

Política de confiança

A política de confiança deve permitir que o serviço de agente do AWS Security Agent assuma essa função:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "your-external-id"
        }
      }
    }
  ]
}
```

Política de permissões

As permissões dependem da arquitetura do aplicativo de destino. Aqui estão alguns exemplos de cenários comuns:

Para aplicativos do API Gateway

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "execute-api:Invoke"
      ],
      "Resource": "arn:aws:execute-api:us-east-1:*:your-api-id/*"
    }
  ]
}
```

Para URLs da função Lambda

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "lambda:InvokeFunctionUrl",
        "lambda:InvokeFunction"
      ],
      "Resource": "arn:aws:lambda:us-east-1:*:function:your-function-name"
    }
  ]
}
```

Para Application Load Balancer com Autenticação Cognito

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Action": [
        "cognito-idp:InitiateAuth",
        "cognito-idp:RespondToAuthChallenge"
      ],
      "Resource": "arn:aws:cognito-idp:us-east-1:*:userpool/your-user-pool-id"
    }
  ]
}
```

> **Important**
>
> Configure as permissões do Actor Role para corresponder aos requisitos de autenticação e autorização do seu aplicativo de destino. A função deve ter o mesmo nível de acesso dos usuários ou serviços que o Security Agent simulará durante o teste de penetração.

Chaves gerenciadas pelo cliente para o AWS Security Agent

Por padrão, o AWS Security Agent criptografa todos os dados em repouso usando chaves AWS de criptografia gerenciadas. Opcionalmente, você pode usar uma chave gerenciada pelo cliente do AWS Key Management Service (AWS KMS) para criptografar seus dados, oferecendo controle total sobre as chaves de criptografia que protegem seus recursos.

AWS O Security Agent oferece suporte a chaves gerenciadas pelo cliente em nível de recursos. Ao criar um recurso de nível superior, como um Espaço do Agente ou uma integração, você pode especificar uma chave KMS para criptografar todos os dados pertencentes a esse recurso e seus sub-recursos. Por exemplo, especificar uma chave gerenciada pelo cliente ao criar um Espaço do Agente criptografa todos os dados associados a esse Espaço do Agente, incluindo configurações do Espaço do Agente, configurações de testes de penetração, tarefas e detalhes de execução, descobertas de segurança, endpoints descobertos e capturas de tela. Você também pode fornecer AWS recursos que já estão criptografados com sua própria chave gerenciada pelo cliente, como buckets S3, grupos de registros de CloudWatch registros ou segredos do Secrets Manager. Para obter as permissões necessárias do KMS, consultethe section called “Permissões KMS necessárias”.

Como funcionam as chaves gerenciadas pelo cliente

AWS O Security Agent usa o chaveiro hierárquico do AWS KMS para criptografar seus dados com chaves gerenciadas pelo cliente. Quando você especifica uma chave KMS durante a criação do recurso, o serviço cria uma chave de ramificação protegida pela sua chave KMS e a armazena em

um repositório de DynamoDB-based chaves da Amazon. Para cada operação de criptografia, o chaveiro hierárquico deriva uma chave de encapsulamento exclusiva da chave de ramificação ativa, que criptografa uma chave de criptografia de dados exclusiva para cada solicitação.

O chaveiro hierárquico oferece os seguintes benefícios:

- Per-resource escopo de criptografia — Cada recurso de nível superior tem sua própria chave de ramificação, portanto, os dados de diferentes recursos são criptografados em chaves separadas.
- Rotação automática de chaves — o AWS Security Agent gira periodicamente as chaves de ramificação. Após a rotação, os novos dados são criptografados com a nova versão da chave de ramificação, enquanto as versões mais antigas são retidas para descriptografar os dados criptografados anteriormente.

Contexto de criptografia

AWS O Security Agent usa contexto de criptografia em todas as operações criptográficas com sua chave KMS. O contexto de criptografia é um conjunto de pares de valores-chave não secretos que fornece dados autenticados adicionais para a operação de criptografia.

A chave de contexto de criptografia segue o formato`aws:securityagent:_<resource-type>_`, onde *`<resource-type>`* é o tipo de recurso que está sendo criptografado (por exemplo, `agent-space` ou`integration`). O valor é o Amazon Resource Name (ARN) do recurso.

> **Note**
>
> Para integrações, a chave de contexto de criptografia inclui o `aws-crypto-ec:` prefixo, resultando no formato. `aws-crypto-ec:aws:securityagent:integration`

Você pode usar o contexto de criptografia para definir a política de chaves do KMS para tipos específicos de recursos. Para obter mais informações, consulte the section called “Permissões KMS necessárias”.

Default encryption (Criptografia padrão)

Quando você cria um recurso sem especificar uma chave gerenciada pelo cliente, o AWS Security Agent criptografa o recurso usando chaves de criptografia AWS gerenciadas fornecidas pelos serviços de armazenamento subjacentes (Amazon DynamoDB e Amazon S3). Nenhuma configuração adicional é necessária para a criptografia padrão.

Chave KMS padrão para aplicativos

Você pode definir uma chave KMS padrão no nível do aplicativo. Quando uma chave KMS padrão é configurada, o AWS Security Agent a usa como alternativa para novos Agent Spaces e integrações quando você não especifica explicitamente uma chave KMS durante a criação do recurso.

Para definir uma chave KMS padrão, especifique o `defaultKmsKeyId` parâmetro ao criar ou atualizar seu aplicativo usando a AWS CLI ou o SDK. O console solicita que você especifique uma chave KMS padrão durante a configuração do aplicativo.

Quando você especifica explicitamente uma chave KMS durante a criação do recurso, ela tem precedência sobre o padrão no nível do aplicativo.

Pré-requisitos

Antes de configurar uma chave gerenciada pelo cliente, preencha os seguintes pré-requisitos:

- Crie uma chave KMS de criptografia simétrica no AWS KMS. A chave deve atender aos seguintes requisitos:
  - Tipo de chave: Simétrico
  - Uso da chave: criptografar e descriptografar
  - Especificação chave: `SYMMETRIC_DEFAULT`
  - Estado da chave: Ativado

  Para obter instruções, consulte Criação de chaves no Guia do desenvolvedor do AWS Key Management Service.
- Configure a política de chaves do KMS e as políticas do IAM para conceder ao AWS Security Agent as permissões necessárias. Para obter detalhes, consulte the section called “Permissões KMS necessárias”.

Permissões KMS necessárias

AWS O Security Agent exige permissões KMS em dois cenários:

- Criptografar dados no AWS Security Agent — Quando você especifica uma chave gerenciada pelo cliente para um Espaço do Agente ou integração, o serviço precisa de permissões para usar essa chave para operações criptográficas em seus dados.

- Uso de CMK-encrypted AWS recursos — Quando você fornece AWS recursos criptografados com uma chave gerenciada pelo cliente (como buckets do S3, grupos de CloudWatch registros de registros ou segredos do Secrets Manager), o serviço precisa de permissões para usar essas chaves para acessar os recursos criptografados.

AWS O Security Agent acessa sua chave KMS usando identidades diferentes, dependendo da operação:

- AWS Operações do Management Console — Quando os administradores criam ou gerenciam recursos no AWS Management Console (por exemplo, criando um Espaço do Agente ou atualizando um aplicativo), o serviço usa a função de administrador para chamar o AWS KMS.
- Operações do aplicativo Web — Quando os usuários do IAM Identity Center acessam dados por meio do aplicativo web do AWS Security Agent (por exemplo, visualizando os resultados do teste de penetração ou iniciando um teste de penetração), o serviço usa a função de aplicativo criada durante a configuração do AWS Security Agent para chamar AWS o KMS.
- Acessando recursos fornecidos pelo cliente — Quando o serviço acessa os AWS recursos fornecidos pelo cliente durante o teste de penetração (como buckets do S3, CloudWatch grupos de registros de registros e segredos do Secrets Manager), ele usa a função de serviço de teste de penetração para chamar o KMS. AWS
- Fluxos de trabalho assíncronos — Para operações em segundo plano que são executadas fora de qualquer sessão do usuário (por exemplo, execução de testes de penetração, processamento de revisão de código e rotação de chaves de ramificação), o serviço usa seu próprio serviço principal (`securityagent.amazonaws.com`) para chamar AWS o KMS em seu nome.

As seções a seguir descrevem as permissões necessárias para cada identidade.

Política de chave

Sua política de chaves do KMS deve conceder permissão ao AWS Security Agent para usar a chave para operações criptográficas. As principais declarações de política necessárias dependem dos tipos de recursos que você planeja criptografar e CMK-encrypted AWS dos recursos que você fornece ao serviço.

Política fundamental para Agent Spaces

A política de chaves a seguir concede ao AWS Security Agent as permissões necessárias para criptografar e descriptografar dados do Agent Space, incluindo resultados de testes de penetração e capturas de tela.

Substitua os seguintes valores de espaço reservado na política:

- *111122223333* — O ID AWS da sua conta
- *MyRole* — A função do IAM que você usa para gerenciar o AWS Security Agent no console
- *MyApplicationRole* — A função do aplicativo criada durante a configuração do AWS Security Agent
- *us-east-1* — A AWS região onde você usa o AWS Security Agent

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKeyMetadataValidationForApplicationAndAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowUseOfHierarchicalKeyringForAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
```

```
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
"arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        },
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowSynchronousDataAccessForAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyApplicationRole"
      },
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
"arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        },
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowAsynchronousDataAccessForAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
```

```
        "kms:Decrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
 "arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        }
      }
    },
    {
      "Sid": "AllowAsynchronousDataAccessForCodeRemediation",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey"
      ],
      "Resource": "*"
    }
  ]
}
```

Important

Para alternar as chaves de ramificação, sua política de chaves do KMS deve conceder `kms:GenerateDataKeyWithoutPlaintextkms:ReEncryptTo`, e `kms:ReEncryptFrom` permissões ao responsável pelo serviço principal do AWS Security Agent (`securityagent.amazonaws.com`), ou a rotação de chaves de ramificação falhará. Para obter mais informações sobre as permissões necessárias, consulte Girar uma chave de ramificação.

Política-chave para integrações

A política de chaves a seguir AWS concede ao Security Agent as permissões necessárias para criptografar e descriptografar dados de integração, incluindo descobertas de revisão de código.

Substitua os seguintes valores de espaço reservado na política:

- *111122223333* — O ID AWS da sua conta
- *MyRole* — A função do IAM que você usa para gerenciar o AWS Security Agent no console
- *us-east-1* — A AWS região onde você usa o AWS Security Agent

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKeyAccessValidationForIntegrations",
      "Effect": "Allow",
      "Principal": {
          "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws-crypto-ec:aws:securityagent:integration":
 "arn:aws:securityagent:us-east-1:111122223333:integration/*"
        },
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowKeyMetadataValidationForIntegrations",
      "Effect": "Allow",
      "Principal": {
```

```
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "AllowAsynchronousDataAccessForIntegrations",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:securityagent:us-east-1:111122223333:integration/*"
        },
        "StringLike": {
          "kms:EncryptionContext:aws-crypto-ec:aws:securityagent:integration":
 "arn:aws:securityagent:us-east-1:111122223333:integration/*"
        }
      }
    }
  ]
}
```

> **Note**
>
> Você pode combinar as principais declarações de política do Agent Space, da integração e do pacote de requisitos de segurança em uma única política de chave se quiser usar a mesma chave KMS para vários tipos de recursos.

Política chave para pacotes de requisitos de segurança

A política de chaves a seguir concede ao AWS Security Agent as permissões necessárias para criptografar e descriptografar os dados do pacote de requisitos de segurança. Os pacotes de requisitos de segurança não usam uma função de aplicativo web. A política usa dois caminhos de acesso:

- Caminho síncrono — Quando você chama a API, o serviço usa suas credenciais encaminhadas para chamar o AWS KMS em seu nome (por meio da condição). `kms:ViaService`
- Caminho assíncrono — Para operações assíncronas em que os pacotes podem ser usados, o serviço usa seu próprio principal de serviço para chamar o KMS diretamente. AWS

Substitua os seguintes valores na política:

- *`111122223333`* — O ID AWS da sua conta
- *`MyRole`* — A função do IAM que você usa para invocar operações do pacote de requisitos de segurança
- *`us-east-1`* — A AWS região onde você usa o AWS Security Agent

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKeyMetadataValidation",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
```

```
      "Sid": "AllowUseOfHierarchicalKeyringForSecurityPacks",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:security-requirement-pack":
 "arn:aws:securityagent:us-east-1:111122223333:security-requirement-pack/*"
        }
      }
    },
    {
      "Sid": "AllowAsynchronousDataAccessForSecurityPacks",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:security-requirement-pack":
 "arn:aws:securityagent:us-east-1:111122223333:security-requirement-pack/*"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:securityagent:us-east-1:111122223333:security-
requirement-pack/*"
        }
```

```
      }
    }
  ]
}
```

A política contém três declarações:

- AllowKeyMetadataValidation— Concede `kms:DescribeKey` para que o AWS Security Agent possa validar se sua chave gerenciada pelo cliente existe, está habilitada e usa criptografia simétrica quando você especifica uma CMK durante a criação do recurso. Usa a `kms:ViaService` condição para garantir que a chamada seja originada por meio do serviço.
- AllowUseOfHierarchicalKeyringForSecurityPacks— Concede `kms:GenerateDataKeyWithoutPlaintext` (cria novas chaves de ramificação), `kms:ReEncryptTo` `kms:ReEncryptFrom` (gira chaves de ramificação existentes) e `kms:Decrypt` (recupera chaves de ramificação existentes) para operações de chaveiro hierárquico. Essas operações usam suas credenciais encaminhadas por meio do caminho síncrono e têm como escopo os recursos do pacote de requisitos de segurança de acordo com a condição do contexto de criptografia.
- AllowAsynchronousDataAccessForSecurityPacks— Concede`kms:GenerateDataKeyWithoutPlaintext`, `kms:Decryptkms:ReEncryptTo`, e `kms:ReEncryptFrom` ao diretor de serviço do AWS Security Agent para operações assíncronas quando não há credenciais de chamador disponíveis.

Diferentemente da política de chaves do Agent Spaces, a política do pacote de requisitos de segurança não inclui um princípio de função de aplicativo web. Os pacotes de requisitos de segurança são acessados por meio do AWS Management Console usando sua função de administrador, não por meio do aplicativo web do AWS Security Agent. Todas as operações síncronas usam a função de chamador único (via`kms:ViaService`).

> **Note**
>
> Se você usar a mesma chave KMS para Agent Spaces e pacotes de requisitos de segurança, poderá combinar as principais declarações de política de ambas as seções em uma única política de chaves.

Política chave para CMK-encrypted AWS recursos

Se você fornecer AWS recursos criptografados com uma chave gerenciada pelo cliente para o AWS Security Agent, deverá atualizar a política de chaves na chave KMS de cada recurso para conceder as permissões necessárias. Isso se aplica aos seguintes recursos:

- Buckets S3 — Se você fornecer recursos de aprendizado de um bucket S3 criptografado com uma chave gerenciada pelo cliente (SSE-KMS), a política de chaves deve permitir que a função de serviço de teste de penetração decifre objetos.
- Segredos do Secrets Manager — Se suas credenciais de teste de penetração estiverem armazenadas em segredos do Secrets Manager criptografados com uma chave gerenciada pelo cliente, a política de chaves deve permitir que a função do serviço de teste de penetração decifre esses segredos. Se o aplicativo web criar segredos em seu nome, a política de chaves também deverá permitir que a função do aplicativo criptografe esses segredos.
- CloudWatch Grupos de registros de registros — se o grupo de CloudWatch registros de registros usado para armazenar registros de execução de testes de penetração for criptografado com uma chave gerenciada pelo cliente, a política de chaves deverá permitir que o CloudWatch Logs valide a chave KMS por meio da função de serviço e permitir que o diretor do serviço de CloudWatch registros execute operações criptográficas.

> **Important**
>
> AWS O Security Agent também pode criar grupos de CloudWatch registros de registros e segredos do Secrets Manager em seu nome. Ao configurar sua política de chaves do KMS, inclua também as declarações correspondentes para esses recursos criados pelo serviço.

As declarações políticas principais a seguir concedem as permissões necessárias para CMK-encrypted os recursos. Inclua somente as instruções que se aplicam à sua configuração. Se um recurso usar a mesma chave KMS que você especificou para o seu Espaço do Agente, adicione essas declarações à política dessa chave. Se um recurso usar uma chave KMS diferente, adicione essas declarações à política dessa chave.

Substitua os seguintes valores de espaço reservado na política:

- *111122223333* — O ID AWS da sua conta

- *MyApplicationRole* — A função do aplicativo criada durante a configuração do AWS Security Agent
- *MyPenTestServiceRole* — A função do serviço de teste de penetração
- *us-east-1* — A AWS região onde você usa o AWS Security Agent

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKmsKeyAccessForCreatingSecrets",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyApplicationRole"
      },
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyDecryptionForS3Objects",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyPenTestServiceRole"
      },
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
```

```
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "s3.*.amazonaws.com",
          "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::YOUR-BUCKET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyDecryptionForSecretsManagerSecrets",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyPenTestServiceRole"
      },
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:YOUR-SECRET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyValidationForCloudWatchLogsLogGroups",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyPenTestServiceRole"
      },
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
```

```
        "StringLike": {
          "kms:ViaService": "logs.*.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyAccessForCloudWatchLogsLogGroups",
      "Effect": "Allow",
      "Principal": {
        "Service": "logs.us-east-1.amazonaws.com"
      },
      "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:GenerateDataKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:logs:arn": "arn:aws:logs:us-
east-1:111122223333:log-group:YOUR-LOG-GROUP-NAME*"
        }
      }
    }
  ]
}
```

Note

- A `AllowKmsKeyDecryptionForS3Objects` declaração é necessária somente se você fornecer recursos de aprendizado de buckets do S3 criptografados com uma chave gerenciada pelo cliente. Para obter mais informações sobre a configuração do SSE-KMS S3, consulte Protegendo dados com. SSE-KMS
- A `AllowKmsKeyDecryptionForSecretsManagerSecrets` declaração é necessária somente se suas credenciais de teste de penetração estiverem armazenadas em segredos do Secrets Manager criptografados com uma chave gerenciada pelo cliente. A função de serviço precisa ter acesso para decifrar segredos durante o teste de penetração. Para

obter mais informações sobre criptografia secreta, consulte Criptografia e descriptografia secretas no Secrets Manager.

- A `AllowKmsKeyValidationForCloudWatchLogsLogGroups` declaração concede a função de serviço `kms:DescribeKey` para que o CloudWatch Logs possa validar a chave KMS por meio da função de serviço ao associá-la a um grupo de registros.
- A `AllowKmsKeyAccessForCreatingSecrets` declaração é necessária se o aplicativo web criar segredos do Secrets Manager em seu nome usando uma chave gerenciada pelo cliente. A função do aplicativo precisa `kms:GenerateDataKey` `kms:Decrypt` criptografar o segredo com sua chave KMS.
- A `AllowKmsKeyAccessForCloudWatchLogsLogGroups` declaração concede ao serviço CloudWatch Logs principal (`logs.us-east-1.amazonaws.com`) as permissões necessárias para criptografar e descriptografar dados de registro. Essa declaração também é necessária se o AWS Security Agent criar um grupo de registros em seu nome quando você não fornece um existente. Para obter mais informações, consulte Criptografar dados de registro em CloudWatch registros usando o AWS KMS.
- Se vários recursos compartilharem a mesma chave KMS, você poderá combinar as declarações em uma única política de chaves.

Função de administrador

Nenhuma política adicional do IAM é necessária para a função de administrador (a função do IAM que você usa para gerenciar o AWS Security Agent no console).

Perfil da aplicação

Além da política de chaves do KMS, anexe a seguinte política do IAM à função do aplicativo especificada durante a configuração do AWS Security Agent. Essa política concede à função permissões para usar suas chaves gerenciadas pelo cliente para criptografar e descriptografar dados do Agent Space e criar segredos no Secrets Manager. AWS

Substitua os seguintes valores de espaço reservado na política:

- *111122223333* — O ID AWS da sua conta
- *us-east-1* — A AWS região onde você usa o AWS Security Agent
- A chave KMS entra `Resource` — Os ARNs das suas chaves KMS

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSynchronousDataAccessForAgentSpaces",
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890cd"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333",
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
 "arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyAccessForCreatingSecrets",
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890cd"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
```

```
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:*"
        }
      }
    }
  ]
}
```

> **Note**
>
> A função do aplicativo é uma função do IAM que você especifica ou que o console cria durante a configuração do AWS Security Agent. O aplicativo web assume essa função para recuperar os registros de execução do teste de penetração e criar segredos do Secrets Manager em seu nome.
>
> - A `AllowKmsKeyAccessForCreatingSecrets` declaração é necessária se você configurar recursos de autenticação para testes de penetração e optar por inserir as credenciais diretamente em vez de especificar um segredo existente. O aplicativo web cria um segredo em seu nome, e a função do aplicativo precisa das permissões nesta declaração para criptografar o segredo com a chave gerenciada pelo cliente especificada para seu Espaço do Agente. Atualize os `Resource` ARNs nesta declaração para que correspondam à chave KMS usada para seu Espaço do Agente.

Perfil de serviço

Durante o teste de penetração, o AWS Security Agent assume a função de serviço de teste de penetração para acessar seus recursos. AWS Se algum desses recursos for criptografado com uma chave gerenciada pelo cliente, você deverá conceder à função de serviço permissões para usar as chaves KMS correspondentes. Isso se aplica aos seguintes recursos:

- Buckets do S3 — Se você fornecer recursos de aprendizado (como documentos de API, modelos de ameaças ou código-fonte) de um bucket do S3 criptografado com uma chave gerenciada pelo cliente, a função de serviço precisará de permissões para descriptografar objetos nesse bucket. Para obter mais informações sobre a configuração do SSE-KMS S3, consulte Protegendo dados com. SSE-KMS
- Segredos do Secrets Manager — Se suas credenciais de teste de penetração estiverem armazenadas em segredos do Secrets Manager criptografados com uma chave gerenciada pelo cliente, a função de serviço precisará de permissões para descriptografar esses segredos. Para

obter mais informações sobre criptografia secreta, consulte Criptografia e descriptografia secretas no Secrets Manager.

- CloudWatch Grupos de registros de registros — se o grupo de CloudWatch registros de registros usado para armazenar registros de execução de testes de penetração for criptografado com uma chave gerenciada pelo cliente (incluindo grupos de registros criados pelo AWS Security Agent em seu nome), a função de serviço precisará de `kms:DescribeKey` permissão para validar a chave. O responsável pelo serviço de CloudWatch registros gerencia a criptografia e a descriptografia reais dos dados de registro, e essas permissões são concedidas por meio da política de chaves (consulte). the section called “Política de chave” Para obter mais informações sobre a criptografia de dados de registro, consulte Criptografar dados de registro em CloudWatch Registros usando o AWS KMS.

> **Important**
>
> Se você usar uma chave KMS diferente da especificada para seu Espaço do Agente para criptografar esses recursos, você deverá conceder permissões à função de serviço para cada chave KMS que proteja um recurso que a função de serviço acessa durante o teste de penetração.

Anexe a seguinte política do IAM à função do serviço de teste de penetração. Inclua somente as instruções que se aplicam à sua configuração.

Substitua os seguintes valores de espaço reservado na política:

- *111122223333* — O ID AWS da sua conta
- *us-east-1* — A AWS região onde você usa o AWS Security Agent
- A chave KMS entra `Resource` — Os ARNs das chaves gerenciadas pelo cliente usadas para criptografar cada recurso

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKmsAccessForEncryptedS3Buckets",
      "Effect": "Allow",
      "Action": [
```

```
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE-S3-KEY-ID"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "s3.*.amazonaws.com",
          "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::YOUR-BUCKET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsAccessForEncryptedSecrets",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE-SECRETS-KEY-ID"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:YOUR-SECRET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyValidationForCloudWatchLogsLogGroups",
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE-CLOUDWATCH-LOGS-KEY-ID"
      ],
```

```
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "logs.*.amazonaws.com"
        }
      }
    }
  ]
}
```

Note

- A `AllowKmsAccessForEncryptedS3Buckets` declaração é necessária somente se você fornecer recursos de aprendizado de buckets do S3 criptografados com uma chave gerenciada pelo cliente. Atualize o `Resource` ARN para corresponder à chave KMS usada para criptografar seu bucket do S3 e atualize o `kms:EncryptionContext:aws:s3:arn` valor para corresponder ao nome do bucket.
- A `AllowKmsAccessForEncryptedSecrets` declaração é necessária somente se suas credenciais de teste de penetração estiverem armazenadas em segredos do Secrets Manager criptografados com uma chave gerenciada pelo cliente. Atualize o `Resource` ARN para corresponder à chave KMS usada para criptografar seus segredos e atualize o `kms:EncryptionContext:SecretARN` valor para corresponder ao seu nome secreto.
- A `AllowKmsKeyValidationForCloudWatchLogsLogGroups` declaração é necessária somente se seu grupo de CloudWatch registros de registros for criptografado com uma chave gerenciada pelo cliente, incluindo grupos de registros criados pelo AWS Security Agent em seu nome. Atualize o `Resource` ARN para que corresponda à chave KMS usada para criptografar seu grupo de registros.
- Se vários recursos compartilharem a mesma chave KMS, você poderá combinar as instruções e listar o ARN da chave compartilhada uma vez no `Resource` campo.

Crie um recurso com uma chave gerenciada pelo cliente

Você pode especificar uma chave gerenciada pelo cliente ao configurar o AWS Security Agent ou ao criar recursos individuais. A chave KMS criptografa todos os dados pertencentes a esse recurso e seus sub-recursos.

Definir uma chave KMS padrão durante a configuração (console)

Você pode configurar uma chave KMS padrão durante a configuração inicial do AWS Security Agent. Essa chave é usada como padrão para novos Agent Spaces e integrações, a menos que você especifique uma chave diferente.

1. Na página Configurar o AWS Security Agent, expanda a seção Criptografia - opcional.
2. Selecione Personalizar configurações de criptografia (avançado).
3. Em Escolha uma chave do AWS KMS, selecione uma chave KMS da sua conta ou insira um ARN da chave KMS.
4. Conclua as etapas de configuração restantes e escolha Configurar o AWS Security Agent.

AWS O Security Agent valida a chave KMS para confirmar que ela existe, está habilitada e usa criptografia simétrica. Se a validação falhar, a configuração não prosseguirá e você receberá uma mensagem de erro.

Crie um espaço de agente com uma chave gerenciada pelo cliente (console)

1. No console do AWS Security Agent, navegue até a página Agent Spaces.
2. Escolha Criar espaço do agente.
3. Insira um nome e uma descrição opcional para o seu Espaço do Agente.
4. Expanda a seção Advanced.
5. Em Criptografia de dados, escolha uma das seguintes opções:
   - Usar chave padrão do aplicativo — Usa a chave KMS padrão configurada durante a configuração do AWS Security Agent. Essa opção é selecionada por padrão se uma chave padrão estiver configurada.
   - Use uma chave diferente — Especifique uma chave KMS diferente para esse Espaço do Agente. Selecione Personalizar configurações de criptografia (avançadas) e, em Escolha uma chave do AWS KMS, selecione uma chave KMS da sua conta ou insira um ARN.
6. Escolha Criar.

Crie uma integração com uma chave gerenciada pelo cliente (console)

1. No console do AWS Security Agent, navegue até a página Integrações.
2. Escolha Adicionar integração e selecione seu provedor (por exemplo, GitHub).

3. Conclua a Etapa 1: Instale e autorize o aplicativo do provedor.
4. Na Etapa 2: Detalhes do registro, insira um nome de registro e defina as configurações do provedor.
5. Na seção Criptografia de dados, selecione Personalizar configurações de criptografia (avançado).
6. Em Escolha uma chave do AWS KMS, selecione uma chave KMS da sua conta ou insira um ARN da chave KMS.
7. Selecione Conectar.

Crie um recurso com uma chave gerenciada pelo cliente (AWS (CLI ou SDK)

Para especificar uma chave gerenciada pelo cliente usando a AWS CLI ou o SDK:

- Inclua o `defaultKmsKeyId` parâmetro ao chamar `CreateApplication` para definir uma chave KMS padrão para seu aplicativo. Essa chave é usada como alternativa ao criar Agent Spaces ou integrações sem uma chave KMS explícita.
- Inclua o `kmsKeyId` parâmetro ao chamar `CreateAgentSpace` ou `CreateIntegration` criptografar um recurso específico. Isso tem precedência sobre o padrão no nível do aplicativo.

Para obter detalhes dos parâmetros, consulte a Referência da API do AWS Security Agent.

Se você não especificar uma chave KMS, o AWS Security Agent usará a chave KMS padrão no nível do aplicativo, se houver uma configurada. Caso contrário, o recurso será criptografado com chaves AWS gerenciadas.

Alternância de chaves

AWS O Security Agent gira automaticamente as chaves de ramificação periodicamente. A rotação da chave de ramificação cria uma nova versão ativa da chave de ramificação, mantendo todas as versões anteriores. Depois da rotação:

- Os novos dados são criptografados com a nova versão da chave de ramificação.
- Os dados criptografados anteriormente permanecem decifráveis usando a versão mais antiga da chave de ramificação.
- Não é necessária uma nova criptografia de dados.

A rotação da chave de ramificação é separada da rotação da chave KMS. Você pode ativar a rotação automática de chaves KMS para sua chave gerenciada pelo cliente de forma independente. Para obter mais informações, consulte Chaves rotativas do KMS no Guia do desenvolvedor do AWS Key Management Service.

Após a rotação da chave de ramificação, há um breve período durante o qual as cópias em cache da chave de ramificação anterior ainda podem ser usadas para novas operações de criptografia. Essa janela é determinada pelo tempo de vida útil do cache interno (TTL) e não afeta a segurança dos seus dados.

Considerações

Lembre-se do seguinte ao usar chaves gerenciadas pelo cliente com o AWS Security Agent:

- As chaves gerenciadas pelo cliente se aplicam somente a novos recursos. Os recursos existentes criados antes de você configurar uma chave gerenciada pelo cliente continuam usando criptografia AWS gerenciada. Não há migração dos dados existentes.
- As chaves gerenciadas pelo cliente são especificadas no momento da criação. Você especifica a chave KMS ao criar um recurso. Você não pode adicionar ou alterar a chave KMS de um recurso existente.
- Várias chaves KMS são suportadas. Você pode usar chaves KMS diferentes para recursos diferentes. Por exemplo, você pode usar uma chave para Agent Spaces de produção e uma chave diferente para Agent Spaces de desenvolvimento.
- Desativar ou excluir uma chave KMS torna os dados inacessíveis. Se você desabilitar ou agendar a exclusão de uma chave KMS, o AWS Security Agent não poderá descriptografar dados criptografados sob essa chave. Os recursos afetados ficam inacessíveis até que a chave seja reativada. A exclusão de uma chave KMS impede permanentemente o acesso a todos os dados criptografados sob essa chave.
- As principais permissões de política são necessárias. Se você remover as permissões necessárias da sua política de chaves do KMS, o AWS Security Agent não poderá acessar dados criptografados. Certifique-se de que a política de chaves conceda permissões para a função de administrador (usada para gerenciar o serviço no AWS console), a função de aplicativo (usada pelo aplicativo web), a função de serviço de teste de penetração (assumida pelos agentes para executar testes de penetração) e o principal de serviço do AWS Security Agent, conforme descrito em. the section called “Permissões KMS necessárias”
- AWS As cotas do KMS se aplicam. As operações criptográficas em sua chave KMS contam para suas cotas de solicitação do AWS KMS. Em uso normal, a arquitetura hierárquica de chaveiros

minimiza as chamadas KMS por meio do cache de chaves de ramificação. Para obter mais informações, consulte Solicitar cotas no Guia do desenvolvedor do AWS Key Management Service.

Solução de problemas

Encontre soluções para erros comuns ao usar o AWS Security Agent.

Acesso negado: Tipo de GitHub conta incorreto selecionado ou nome incorreto da organização especificado

- Você instalou o aplicativo AWS Security Agent na GitHub organização desejada, mas definiu incorretamente o como `User` em vez `GitHub Account Type` de `Organization`
- Você instalou o aplicativo AWS Security Agent na GitHub organização desejada e definiu corretamente o `GitHub Account Type` como`Organization`, mas deixou o `Organization Name` campo em branco ou inseriu um nome de organização incorreto que não corresponde à organização em que você instalou o aplicativo.

Solução:

1. Acesse GitHub e desinstale o aplicativo da organização. Para obter mais informações, consulte the section called “Etapa 1: Desinstalar o GitHub aplicativo AWS Security Agent do GitHub”.
2. Volte para a página de integrações e reinicie o processo de integração clicando em`Add Integrations`, instale e autorize o aplicativo na GitHub organização desejada novamente.
3. Selecione na `Organization` lista suspensa de `GitHub account type`
4. Certifique-se de `Organization Name` que sua entrada seja EXATAMENTE a mesma em que você instalou o aplicativo.
5. Escolha Connect para criar a integração GitHub da sua organização.

Acesso negado: permissões insuficientes para instalar o GitHub aplicativo na organização

Ao tentar instalar o aplicativo AWS Security Agent na GitHub organização desejada, você verá duas mensagens diferentes no botão na página de instalação.

- Uma organização `Member` verá `Authorize & Request`
- Uma organização `Owner` verá `Install & Authorize`

Você pode verificar se você faz parte `Member` ou faz `Owner` parte da GitHub organização seguindo as etapas abaixo.

1. Acesse github.com
2. Escolha seu perfil no site
3. Navegue até `Organizations` o menu suspenso e escolha-o
4. Encontre a organização na qual você deseja instalar o AWS Security Agent na lista de organizações. Ela especificará se você é um `Member` ou está ao `Owner` lado do nome da organização.

Soluções possíveis:

- Peça a um proprietário que aprove sua solicitação de instalação ANTES de tentar criar a integração
- Peça a um proprietário que atualize sua função na GitHub organização de a `Member` para `Owner` e reinicie o processo de integração novamente

O agente não consegue se conectar ao endpoint durante um teste de penetração

Se o agente do teste de penetração não conseguir fazer chamadas para o URL de destino configurado ou não conseguir navegar com êxito pelo endpoint de destino:

- Se seu endpoint fizer chamadas para domínios fora do URL de destino configurado, verifique se os domínios adicionais foram adicionados como URLs acessíveis em sua configuração de pentest
- No momento, o teste de penetração está disponível apenas para HTTP/HTTPS endpoints que atendem tráfego nas portas 80 ou 443
- Se você tiver um WAF configurado, verifique se o WAF não está bloqueando o tráfego do teste de penetração. Você pode permitir o tráfego de teste de penetração na lista de permissões por `User-Agent` cabeçalho, que será definido como padrão `securityagent`

Obter ajuda adicional

Se você continuar enfrentando problemas depois de tentar estas etapas de solução de problemas:

- Verifique a página de status do serviço AWS Security Agent
- Entre em contato com o AWS Support para obter ajuda com problemas complexos de configuração

Para usuários e desenvolvedores (aplicativo web e IDE)

Execute análises de design, modelos de ameaças, análises de código e testes de penetração no aplicativo Web, em um IDE ou em um provedor de origem conectado e, em seguida, analise e corrija as descobertas.

Faça login no aplicativo web AWS Security Agent

Você deve fazer login no AWS Security Agent Web App para concluir tarefas como:

- Realizar uma revisão do projeto
- Faça um teste de penetração

Métodos de acesso

O AWS Security Agent fornece métodos diferentes para acessar o aplicativo web, dependendo de como sua organização configurou o acesso durante a configuração inicial e se você tem acesso ao console da AWS.

IAM Identity Center (SSO) — Se sua organização habilitou o IAM Identity Center, você pode fazer login usando suas credenciais de SSO. Você deve estar atribuído a pelo menos um espaço de agente no IAM Identity Center antes de poder acessar o aplicativo web.

Acesso de administrador (IAM) — Se você tiver acesso ao AWS Console com as permissões apropriadas, poderá iniciar o aplicativo web por meio do link de acesso do administrador em qualquer página do Agent Space. Esse método fornece autenticação por meio de sua sessão de console existente.

> **Note**
>
> O método de acesso principal que sua organização usa foi determinado durante a configuração inicial do AWS Security Agent. Para obter informações sobre como configurar o acesso e as atribuições do usuário, consulte. the section called “Conceda aos usuários acesso ao aplicativo web AWS Security Agent”

Faça login com o IAM Identity Center (SSO)

Se sua organização configurou o acesso ao IAM Identity Center, você pode fazer login no aplicativo web usando suas credenciais de SSO por meio de vários pontos de entrada.

Pré-requisitos

- Você foi atribuído a pelo menos um espaço de agente no IAM Identity Center
- Você tem suas credenciais de SSO do IAM Identity Center

Acesse o aplicativo web

Escolha um dos seguintes métodos com base em suas necessidades:

Para ver todos os Agent Spaces aos quais você está atribuído:

1. No console do AWS Security Agent, navegue até Configurações.
2. Localize o domínio do aplicativo Web e copie o URL.

> **Tip**
>
> Marque esse URL para facilitar o acesso. Esse é o ponto de entrada universal para visualizar todos os Agent Spaces aos quais você está atribuído.

3. Navegue até o URL do aplicativo web.
4. Insira suas credenciais do IAM Identity Center quando solicitado.
5. Após a autenticação, você verá uma lista de todos os Agent Spaces aos quais você está atribuído.
6. Selecione o Espaço do Agente no qual você deseja trabalhar.

Para acessar diretamente um espaço de agente específico (requer acesso ao AWS Console):

1. Faça login no AWS Management Console.
2. Navegue até o console do AWS Security Agent.
3. Navegue até o Espaço do Agente que você deseja acessar.
4. Escolha uma das seguintes opções:
   - Botão de inicialização do aplicativo web na página de visão geral do Agent Space

- Botão de lançamento do aplicativo web na seção Revisão de código
- Inicie o botão do aplicativo web na seção Teste de penetração

5. Insira suas credenciais do IAM Identity Center quando solicitado.
6. Após a autenticação, você será direcionado diretamente para esse Espaço do Agente no aplicativo web.

> **Note**
>
> Se você não tiver acesso ao AWS Console, use o domínio do aplicativo web na página Configurações para acessar todos os seus Agent Spaces atribuídos.

Faça login com acesso de administrador (IAM)

Se você tiver acesso ao console da AWS com as permissões apropriadas do AWS Security Agent, poderá iniciar o aplicativo web por meio do link de acesso do administrador com autenticação automática.

Pré-requisitos

- Você está conectado ao AWS Management Console
- Você tem as permissões apropriadas para acessar os recursos do AWS Security Agent

Acesse o aplicativo web

Escolha um dos seguintes métodos:

Para acessar um espaço de agente específico:

1. Faça login no AWS Management Console.
2. Navegue até o console do AWS Security Agent.
3. Navegue até o Espaço do Agente que você deseja acessar.
4. Escolha o botão de acesso de administrador na página de visão geral do Agent Space.
5. O aplicativo web é aberto em uma nova guia com autenticação automática para esse Espaço do Agente.

> **Note**
>
> O botão de acesso de administrador está sempre disponível para usuários com acesso ao AWS Console, independentemente de sua organização usar o IAM Identity Center como o principal método de acesso.

Crie uma revisão de design

Avalie seus documentos de design em relação aos requisitos de segurança da organização fazendo o upload de arquivos para análise do AWS Security Agent. As análises de design ajudam a identificar problemas de segurança no início do ciclo de vida do desenvolvimento, permitindo que você resolva problemas de arquitetura quando for mais econômico resolvê-los.

O AWS Security Agent analisa seus documentos de design em relação aos requisitos de segurança da sua organização, fornecendo descobertas de segurança detalhadas para melhorar a postura de segurança antes do início da implementação.

Neste procedimento, você criará uma revisão de design carregando arquivos de design para análise.

Pré-requisitos

Antes de começar, verifique se você tem:

- Acesso ao aplicativo web do AWS Security Agent
- Crie documentos prontos para upload (DOC, DOCX, JPEG, MD, PDF, PNG e TXT)
- Cada arquivo deve ter 2 MB ou menos, com um total combinado de 6 MB em todos os arquivos
- Compreender quais requisitos de segurança estão habilitados para sua organização

Etapa 1: comece a criar uma revisão de design

Navegue até a página de criação da revisão de design no Agent Web App.

1. Faça login no aplicativo web do AWS Security Agent.
2. Navegue até a seção Revisões de design.
3. Escolha Criar revisão de design.

> **Tip**
>
> Você pode visualizar os requisitos de segurança habilitados da sua organização navegando até a página de requisitos de segurança no console do AWS Security Agent. Selecione qualquer requisito ativado para ver seus detalhes. Esses requisitos são usados para analisar seus arquivos de design.

Etapa 2: Nomeie sua revisão de design

Forneça um nome descritivo que ajude a identificar a finalidade e o escopo dessa revisão de design.

1. Na seção Nome da revisão de design, localize o campo Nome.
2. Insira um nome descritivo para sua revisão de design.

> **Note**
>
> O nome deve identificar claramente o projeto, recurso ou componente que está sendo revisado. Máximo de 80 caracteres.

Etapa 3: fazer upload de arquivos de design

Faça upload dos documentos de design que você deseja que o AWS Security Agent analise para verificar a conformidade de segurança.

1. Na seção Arquivos a serem revisados, revise os requisitos do arquivo:

> **Important**
>
> No máximo 5 arquivos podem ser enviados por revisão de design. Cada arquivo deve ter 2 MB ou menos, com um total combinado de 6 MB em todos os arquivos. Formatos suportados: DOC, DOCX, JPEG, MD, PDF, PNG e TXT.

2. Faça upload de seus arquivos usando um destes métodos:
   a. Arrastar e soltar — Arraste os arquivos diretamente para a área da zona suspensa de arquivos
   b. Navegar — Use o botão Escolher arquivos para procurar e selecionar arquivos do seu computador

3. Verifique se todos os arquivos necessários foram enviados.

> **Tip**
>
> Para obter melhores resultados, inclua diagramas de arquitetura, especificações de projeto e documentação técnica que descrevam os componentes e fluxos de dados relevantes para a segurança do seu sistema.

Etapa 4: iniciar a revisão do projeto

Depois de configurar todas as informações necessárias, inicie a análise de segurança de seus documentos de design.

1. Revise todos os arquivos e configurações enviados para garantir a precisão.
2. Escolha Iniciar revisão do design.
3. O AWS Security Agent analisa seus documentos de design em relação aos requisitos de segurança habilitados.

> **Note**
>
> O processo de revisão do design geralmente é concluído em minutos, dependendo do número e do tamanho dos arquivos enviados. Você recebe descobertas de segurança com base nos requisitos de segurança da sua organização.

Próximas etapas

Depois de iniciar sua análise de design:

- Monitore o progresso da revisão no Agent Web App
- Analise descobertas de segurança
- Compartilhe as descobertas com sua equipe de desenvolvimento
- Aborde as descobertas de segurança identificadas em seu projeto
- Atualize os documentos de design e reenvie, se necessário

Revise os resultados de uma revisão de design

As descobertas da análise ajudam você a entender quais requisitos de segurança são atendidos, quais precisam de atenção e quais ações devem ser tomadas para melhorar a postura de segurança do seu projeto antes do início da implementação.

Neste procedimento, você aprenderá como acessar, filtrar e interpretar as descobertas da análise de projeto para abordar as descobertas de segurança de forma eficaz.

Pré-requisitos

Antes de começar, verifique se você tem:

- Uma revisão completa do projeto
- Acesso ao aplicativo web do AWS Security Agent
- Familiaridade com os requisitos de segurança habilitados da sua organização

Etapa 1: acessar a revisão do design

Navegue até a revisão do projeto para ver as descobertas e as informações resumidas.

1. Faça login no aplicativo web do AWS Security Agent.
2. Navegue até a seção Revisões de design.
3. Selecione a revisão do design que você deseja examinar na lista.

> **Tip**
>
> A página de detalhes da revisão do design exibe um resumo do status da revisão, da data de conclusão e do número de arquivos revisados.

Etapa 2: Revise o resumo das descobertas

Examine o resumo de alto nível para entender a postura geral de segurança do seu design.

1. Localize a seção Resumo.
2. Analise a contagem de cada categoria de status de conformidade: Compatível Non-compliant, Dados insuficientes e Não aplicável.

> **Note**
>
> O resumo fornece contagens para cada tipo de status, ajudando você a avaliar rapidamente o número de descobertas que requerem atenção. Para obter explicações detalhadas de cada status, consulte a Etapa 4.

Etapa 3: filtrar e navegar pelas descobertas

Use os recursos de filtragem e pesquisa para se concentrar em descobertas específicas ou status de conformidade.

1. Na seção Revisar descobertas, localize os controles de filtro.
2. Para filtrar por status:
   a. Escolha o menu suspenso de status.
   b. Selecione um status de conformidade específico para visualizar somente as descobertas com esse status.
3. Para pesquisar requisitos de segurança específicos:
   a. Insira as palavras-chave no campo de pesquisa.
   b. Os resultados são atualizados automaticamente à medida que você digita.
4. Use os controles de paginação para navegar por várias páginas de descobertas.

> **Tip**
>
> Filtre primeiro por Non-compliantstatus para priorizar as descobertas que exigem atenção imediata em seu design.

Etapa 4: Entenda os status de conformidade

Cada descoberta exibe um status de conformidade que indica como seu projeto aborda um requisito de segurança específico:

- Compatível — Seu design atende aos requisitos de segurança com base na análise
- Non-compliant— Seu design viola ou aborda inadequadamente o requisito de segurança

- Dados insuficientes — Os arquivos enviados não têm informações suficientes para determinar a conformidade
- Não aplicável — O requisito de segurança não se aplica ao design do seu sistema

> **Important**
>
> Concentre-se em Non-compliantstatus de dados insuficientes, pois eles exigem ação. Aborde as descobertas não compatíveis atualizando seu projeto e resolva as descobertas de dados insuficientes fazendo o upload de documentação adicional do projeto.

Etapa 5: ver os detalhes da descoberta

Selecione descobertas individuais para ver a justificativa detalhada e a orientação de remediação.

1. Na tabela de descobertas, selecione o nome de um requisito de segurança.
2. Analise os detalhes da descoberta, que incluem:
   - O requisito de segurança específico que está sendo avaliado
   - Um comentário explicando por que a descoberta recebeu seu status de conformidade, incluindo detalhes específicos sobre o que está faltando ou não está em conformidade
   - Orientação de remediação recomendada para abordar a descoberta
   - Links para a documentação interna ou os padrões da sua organização para o requisito de segurança

> **Note**
>
> O comentário explica o raciocínio do AWS Security Agent com detalhes específicos. Para resultados de dados insuficientes, o comentário identifica quais informações estão faltando, como “Os documentos de design não mencionam mecanismos de autenticação” ou “Nenhuma informação encontrada sobre criptografia de dados em repouso”.

Etapa 6: abordar as descobertas

Tome medidas com base em descobertas que exijam atenção para melhorar a postura de segurança do seu projeto.

Para Non-compliantdescobertas:

1. Revise a orientação de remediação recomendada.
2. Revise qualquer documentação interna vinculada para obter mais contexto.
3. Atualize seus documentos de design para atender aos requisitos de segurança.
4. Documente as alterações feitas para futura referência.

Para resultados de dados insuficientes:

1. Leia o comentário com atenção para entender quais informações específicas estão faltando.
2. Crie ou atualize documentos de design com os detalhes que faltam.
3. Prepare os arquivos atualizados para reenvio.

Próximas etapas

Depois de analisar suas descobertas de design:

- Baixe o relatório de descobertas como um arquivo CSV para compartilhar com sua equipe
- Atualize os documentos de design para abordar descobertas não compatíveis
- Crie documentação adicional para descobertas de dados insuficientes
- Compartilhe as descobertas com sua equipe de desenvolvimento para discussão
- Clone essa revisão de design para criar uma nova revisão com os documentos originais pré-carregados, permitindo que você atualize o nome e execute a análise novamente para verificar melhorias
- Prossiga com a implementação de projetos que atendam aos requisitos de conformidade

Para obter mais informações sobre o gerenciamento dos requisitos de segurança, consultethe section called “Gerencie os requisitos de segurança”.

Para obter mais informações sobre a criação de revisões de design, consultethe section called “Crie uma revisão de design”.

Crie um modelo de ameaça

Crie modelos de ameaças no aplicativo web do AWS Security Agent para analisar a arquitetura do seu aplicativo e identificar ameaças à segurança. Um modelo de ameaça produz duas saídas principais: uma visão geral do sistema que descreve como seu sistema é construído e se comporta, e um conjunto de ameaças que descreve como o sistema pode ser atacado, cada uma com um nível de severidade, classificação STRIDE e recomendações acionáveis.

Um modelo de ameaça aceita dois tipos de entradas, e você pode fornecer uma ou ambas:

- Documentos de escopo — documentos de design técnico, especificações de API ou documentação de arquitetura que definem em que o modelo de ameaças se concentra. Quando fornecido, o agente direciona sua análise para o contexto descrito nesses documentos. Você pode fazer upload de arquivos (DOC, DOCX, JPEG, MD, PDF, PNG, TXT), fornecer URIs do S3 ou conectar páginas do Confluence por meio de uma integração.
- Fontes — Código-fonte de repositórios conectados (GitHub, GitHub Enterprise Server, GitLab GitLab Self-Managed, ou Bitbucket) ou de locais do S3. O AWS Security Agent lê o código-fonte para entender seu sistema atual. Quando combinado com documentos de escopo, o código-fonte fornece contexto sobre a implementação atual.

Neste procedimento, você cria um modelo de ameaça configurando suas entradas e permissões e, em seguida, inicia uma execução.

Pré-requisitos

Antes de começar, verifique se você tem:

- Acesso ao aplicativo web do AWS Security Agent
- Pelo menos um dos seguintes:
  - Um repositório conectado (GitHub, GitHub Enterprise Server, GitLab GitLab Self-Managed, ou Bitbucket) para usar como fonte (consulte) the section called “Ative a modelagem de ameaças”
  - Um bucket S3 contendo código-fonte
  - Crie documentos para serem carregados como documentos de escopo ou como uma integração com o Confluence

> **Tip**
>
> Se você já tem repositórios ou buckets S3 conectados ao seu Agent Space (por exemplo, por meio de análise de código ou configuração de teste de penetração), você pode criar um modelo de ameaça imediatamente — nenhuma configuração adicional é necessária.

Como o AWS Security Agent executa um modelo de ameaça

As entradas que você fornece determinam como o AWS Security Agent executa o modelo de ameaça. Há três cenários.

Entradas que você fornece	Como o AWS Security Agent executa o modelo de ameaças
Somente fontes (código-fonte)	O AWS Security Agent analisa o código-fonte para entender a estrutura do seu aplicativo, classifica componentes, extrai fluxos de dados, cria um modelo de ameaça e identifica ameaças com base em como o sistema é realmente implementado.
Somente documentos de escopo (documentos de design)	O AWS Security Agent executa um modelo de ameaça focado no design descrito em seus documentos. Ele identifica ameaças com base na arquitetura, nos fluxos de dados e nos componentes descritos nos documentos do escopo — útil para modelar um design de ameaças antes que qualquer código exista.
Documentos de fontes e escopo	O AWS Security Agent define o modelo de ameaça de acordo com o contexto descrito na documentação do escopo (por exemplo, um documento de design para um novo recurso). Ele usa o código-fonte para entender seu sistema existente e, em seguida, modela o design de ameaças descrito na documentação do escopo, independentemente de esse design já estar implementado ou não.

Acesse a página de modelos de ameaças

Navegue até a seção de modelagem de ameaças no aplicativo web.

1. Faça login no aplicativo web do AWS Security Agent.
2. Na barra lateral esquerda, escolha Modelos de ameaças.
3. Você vê uma lista dos modelos de ameaças existentes com seu título, status de execução mais recente e contagens de ameaças por gravidade.

Crie um modelo de ameaça

Configure um novo modelo de ameaça configurando suas entradas e permissões.

1. Na página Modelos de ameaças, escolha Criar modelo de ameaça.

Configurar detalhes do modelo de ameaça

Forneça um título para seu modelo de ameaça.

1. No campo Título, insira um nome descritivo para seu modelo de ameaça.

> **Tip**
>
> Use um nome que identifique o aplicativo ou serviço que está sendo modelado. Por exemplo, “billing-service” ou “checkout-api”.

Adicionar documentos de escopo

Forneça os documentos de design técnico que definem o foco do modelo de ameaças, como documentação de arquitetura, especificações de API ou propostas de design. Quando os documentos de escopo são fornecidos, o agente define sua análise de acordo com o contexto descrito nesses documentos. Os documentos de escopo são opcionais se você fornecer fontes.

1. Na seção Documentos de escopo, adicione documentos usando um ou mais dos seguintes métodos:
   - Carregar arquivos — Carregue documentos de design diretamente (DOC, DOCX, JPEG, MD, PDF, PNG ou TXT).
   - URIs do S3 — Insira os URIs do S3 apontando para documentos armazenados nos buckets do S3 conectados.

- Páginas do Confluence — Se você tiver uma integração do Confluence conectada ao seu Espaço do Agente, selecione páginas do seu espaço de trabalho do Confluence.

> **Tip**
>
> Para obter melhores resultados, inclua diagramas de arquitetura, especificações de API e documentação técnica que descrevam os componentes, os fluxos de dados e os limites de confiança do seu sistema.

Adicionar fontes

Selecione o código-fonte que o AWS Security Agent usa para entender seu sistema atual. Quando combinado com documentos de escopo, o código-fonte fornece contexto sobre a implementação atual — o agente o usa para entender o que já existe. As fontes são opcionais se você fornecer documentos de escopo.

1. Na seção Fontes, adicione o código-fonte usando um ou mais dos seguintes métodos:

Repositórios integrados

Selecione entre os repositórios conectados ao seu Espaço do Agente (GitHub GitLab GitLab Self-Managed, GitHub Enterprise Server ou Bitbucket).

1. Marque a caixa de seleção ao lado de cada repositório que você deseja incluir como fonte.
2. Use o campo de pesquisa para encontrar repositórios específicos por nome.

> **Note**
>
> Somente repositórios conectados ao seu Espaço do Agente aparecem aqui. Para adicionar mais repositórios, peça ao administrador que atualize a configuração do Espaço do Agente.

Fontes do S3

Adicione URIs do S3 apontando para o código-fonte armazenado nos buckets do S3 conectados.

1. Insira o URI do S3 para cada local do código-fonte que você deseja incluir.

Configurar recursos da AWS

Selecione a função de serviço do IAM e o grupo de CloudWatch registros opcional para esse modelo de ameaça.

1. No menu suspenso Função de serviço, selecione a função do IAM em suas funções de serviço configuradas.

> **Note**
>
> A função de serviço deve ter permissões para acessar seu código-fonte no S3 e gravar nos CloudWatch registros. As funções de serviço são configuradas durante a configuração do Agent Space no AWS Management Console.

2. (Opcional) No menu suspenso Grupo de registros, selecione um grupo de registros para armazenar os CloudWatch registros de execução do modelo de ameaça.

Crie o modelo de ameaça

1. Como analisar a sua configuração do .
2. Escolha Criar modelo de ameaça.

Você é redirecionado para a página de detalhes do modelo de ameaça, onde pode iniciar uma corrida.

Execute um modelo de ameaça

Depois de criar um modelo de ameaça, inicie uma execução para iniciar a análise.

1. Na página de detalhes do modelo de ameaça, escolha Iniciar execução.
2. O AWS Security Agent começa a analisar suas fontes e documentos de escopo.

Você também pode iniciar uma execução na página de lista de modelos de ameaças escolhendo Iniciar execução ao lado do modelo de ameaça que você deseja executar.

Monitore a execução de um modelo de ameaça

Acompanhe o progresso do seu modelo de ameaça à medida que ele é executado.

Run status (Status da execução)

O cabeçalho da página de execução exibe um indicador de status:

- Em andamento — O agente do modelo de ameaças está em execução.
- Parando — Foi solicitado um cancelamento; o agente está concluindo sua tarefa atual antes de parar.
- Concluído — A execução foi concluída com sucesso.
- Falha — A execução encontrou um erro. Uma mensagem de erro é exibida com detalhes. Inicie uma nova execução depois de resolver o problema.
- Interrompido — A execução foi cancelada pelo usuário. Todas as ameaças geradas antes da parada são preservadas.

Executar guias de página

Na página de detalhes da execução, navegue entre as guias:

- Visão geral — Visualize o resumo da execução (ID do trabalho, horário de início, status, duração) com um gráfico circular do nível de gravidade mostrando a divisão das ameaças por gravidade (crítica, alta, média, baixa). Abaixo do resumo, veja um gráfico de barras das categorias de ameaças do STRIDE mostrando quantas ameaças se enquadram em cada categoria do STRIDE. Depois que a execução for concluída, visualize a visão geral do sistema produzida pelo agente.
- Configuração — Visualize as fontes, documentos e permissões (função de serviço, grupo de CloudWatch registros) que foram usados para essa execução.
- Preflight — Visualize o status das verificações de preflight que são executadas antes do início da análise de ameaças, incluindo a configuração da infraestrutura de registro, a validação do acesso à fonte do S3 (quando aplicável) e a configuração do ambiente de teste. Uma barra de progresso mostra quantas verificações foram concluídas.
- Registros — Visualize uma lista filtrável das tarefas que o agente executou durante a execução. Selecione qualquer tarefa para ver sua saída de registro detalhada em um painel lateral.
- Ameaças — Visualize as ameaças identificadas durante a execução (consultethe section called “Analise as ameaças a partir de um modelo de ameaça”).

Histórico de execução

Cada modelo de ameaça mantém um histórico de todas as execuções. Na página de detalhes do modelo de ameaças:

- A tabela de execuções lista todas as execuções com sua hora de início, status, duração, contagem de ameaças e ID.
- Escolha o link do horário de início de qualquer corrida para ver todos os detalhes.

> **Tip**
>
> Re-run o mesmo modelo de ameaça depois de atualizar seu código-fonte ou revisar seus documentos de escopo para ver como a visão geral do sistema e as ameaças mudam com o tempo.

Edite um modelo de ameaça

Para modificar a configuração do seu modelo de ameaça:

1. Na página de detalhes do modelo de ameaça, escolha Editar.
2. Atualize o título, as fontes, os documentos do escopo ou os recursos da AWS conforme necessário.
3. Escolha Salvar alterações.

> **Note**
>
> A edição de um modelo de ameaça não afeta as execuções concluídas anteriormente. Eles preservam o instantâneo de configuração usado no momento em que foram executados.

Excluir um modelo de ameaça

Para excluir um modelo de ameaça e todas as execuções e ameaças associadas:

1. Na página de detalhes do modelo de ameaça, abra o menu de ações e escolha Excluir.

2. Confirme a exclusão.

> **Important**
>
> Se uma execução estiver em andamento no momento, você deverá interrompê-la antes de excluir o modelo de ameaça.

Próximas etapas

Depois de executar um modelo de ameaça:

- Analise a visão geral do sistema e as ameaças (consultethe section called “Analise as ameaças a partir de um modelo de ameaça”)
- Re-run o modelo de ameaça após fazer alterações para verificar se as ameaças foram tratadas
- Ajuste suas fontes e documentos de escopo à medida que seu aplicativo evolui

Analise as ameaças a partir de um modelo de ameaça

Após a conclusão da execução do modelo de ameaças, analise a visão geral do sistema e as ameaças para entender como seu aplicativo pode ser atacado e o que fazer a respeito. A visão geral do sistema é um documento abrangente que descreve a arquitetura, os limites de confiança, os fluxos de dados e a postura de segurança do seu aplicativo. Cada ameaça inclui uma declaração, nível de gravidade, classificação STRIDE, ativos afetados e uma recomendação para abordá-la.

Pré-requisitos

Antes de começar, verifique se você tem:

- Uma execução completa do modelo de ameaças
- Acesso ao aplicativo web do AWS Security Agent

Etapa 1: acessar a execução do modelo de ameaça

Navegue até a execução completa do modelo de ameaças.

1. Faça login no aplicativo web do AWS Security Agent.

2. Na barra lateral esquerda, escolha Modelos de ameaças.
3. Selecione o modelo de ameaça que você deseja examinar.
4. Na tabela de execuções, selecione a execução concluída escolhendo seu link de horário de início.

Etapa 2: Revise a visão geral do sistema

A visão geral do sistema é uma descrição abrangente de como o AWS Security Agent entende seu sistema. É um documento estruturado que pode incluir:

- Propósito — O que o sistema faz e a quem serve.
- Capacidades — Principais funcionalidades que o sistema fornece.
- Intenção do projeto — A alteração do design ou o recurso que está sendo modelado pela ameaça (quando os documentos do escopo são fornecidos).
- Arquitetura — Como o sistema é construído, incluindo padrões de implantação e protocolos de comunicação.
- Componentes — Uma tabela de componentes do sistema com suas finalidades e principais interações.
- Limites de confiança — Onde os contextos de segurança mudam, incluindo quais proteções existem em cada cruzamento.
- Fluxos de dados — descrições detalhadas de como os dados se movem pelo sistema, incluindo protocolos, credenciais e proteções em cada etapa.
- Postura de segurança — Mecanismos atuais de autenticação, criptografia e controle de acesso.
- Ativos confidenciais — Dados e credenciais que exigem proteção, com sua classificação e pontos de exposição.
- Principais suposições — Security-relevant suposições que o agente fez sobre o sistema.

Para revisar a visão geral do sistema:

1. Selecione a guia Visão geral.
2. Revise a seção Resumo da execução, que mostra o ID do trabalho, a hora de início, o status e a duração.
3. Analise o gráfico do nível de severidade, que mostra uma divisão das ameaças por gravidade (crítica, alta, média, baixa) com contagens e porcentagens.

4. Analise o gráfico de categorias de ameaças, que mostra quantas ameaças se enquadram em cada categoria STRIDE (falsificação, adulteração, repúdio, divulgação de informações, negação de serviço, elevação de privilégio).
5. Na seção Visão geral do sistema, leia a análise completa do agente.

> **Tip**
>
> Se a visão geral do sistema não refletir seu sistema com precisão, refine suas entradas — adicione repositórios relevantes como fontes ou faça upload de documentos de escopo mais completos — e execute o modelo de ameaça novamente.

Etapa 3: Analise as ameaças

Navegue até a guia Ameaças para ver todas as ameaças identificadas durante a execução.

1. Selecione a guia Ameaças.
2. As ameaças são exibidas como uma lista com cada cartão mostrando o título, o distintivo de gravidade, o status e a data da última atualização da ameaça. Você pode filtrar ameaças por gravidade, status ou pesquisar por título.
3. Selecione uma ameaça na lista para ver seus detalhes completos no painel direito.

Gravidade da ameaça

Cada ameaça recebe um nível de gravidade:

- Crítico — requer ação imediata; a exploração pode levar ao comprometimento total do sistema.
- Alto — requer atenção imediata; a exploração pode resultar em um impacto significativo na segurança.
- Médio — Deve ser tratado em um prazo razoável; contribui para o risco geral de segurança.
- Baixo — Pode ser tratado como parte da manutenção regular; risco imediato mínimo.

Detalhes da ameaça

Selecione uma ameaça para ver seus detalhes no painel direito. Os detalhes estão organizados nas seguintes seções:

Linha de metadados:

- Gravidade — O nível de risco atribuído pelo agente (crítico, alto, médio ou baixo).
- Categorias STRIDE — A classificação de ameaças: falsificação, adulteração, repúdio, divulgação de informações, negação de serviço ou elevação de privilégio.
- Criado em — Quando a ameaça foi identificada.
- Última atualização — Quando a ameaça foi modificada pela última vez.

Seção de descrição:

- Declaração — Uma descrição em linguagem natural da ameaça: o que a fonte da ameaça pode fazer, qual é o impacto e quais condições o possibilitam.
- Fonte — O ator ou a origem da ameaça (por exemplo, “usuário autenticado” ou “atacante externo”).
- Ação — O que a fonte da ameaça pode fazer (por exemplo, “injetar consultas SQL no parâmetro de pesquisa”).
- Impacto — A consequência direta da ação da ameaça (por exemplo, “acesso não autorizado aos registros do cliente”).

Seção de referências:

- Ativos afetados — ativos específicos afetados pela ameaça (por exemplo, “registros de pagamento do cliente” ou “tabela do DynamoDB”).
- Pré-requisitos — Condições que devem ser verdadeiras para que a ameaça seja explorável.
- Metas afetadas — Metas de segurança afetadas: confidencialidade, integridade, disponibilidade, autorização, autenticação ou. Non-repudiation
- Evidências — Caminhos de arquivo de código-fonte que suportam a ameaça, vinculados a arquivos específicos em seu repositório.
- Âncora — Uma referência de código que vincula a ameaça a um nó específico em seu código-fonte.

Seção de recomendação:

- Recomendação — orientação prática para lidar com a ameaça.

Etapa 4: criar uma ameaça manualmente

Você pode adicionar ameaças que o agente não identificou, por exemplo, ameaças descobertas durante a análise manual ou de fontes externas.

1. Na guia Ameaças de uma execução concluída, escolha Criar ameaça.
2. Preencha os detalhes da ameaça:
   - Declaração — Uma descrição em linguagem natural da ameaça.
   - Gravidade — Selecione Crítica, Alta, Média ou Baixa.
   - Fonte — O ator ou a origem da ameaça.
   - Pré-requisitos — Condições necessárias para que a ameaça seja explorável.
   - Ação — O que a fonte da ameaça pode fazer.
   - Impacto — A consequência direta da ação da ameaça.
   - Ativos afetados — ativos específicos afetados (separados por vírgula).
   - Metas de segurança afetadas — selecione entre Confidencialidade, Integridade, Disponibilidade, Autorização, Autenticação ou. Non-repudiation
   - Categorias STRIDE — Selecione as categorias aplicáveis.
   - Recomendação — Orientação para lidar com a ameaça.
3. Escolha Criar.

A ameaça criada manualmente aparece na lista de ameaças junto com as ameaças geradas pelo agente.

Etapa 5: editar e fazer a triagem de ameaças

Ao analisar as ameaças, você pode editar seus detalhes e atualizar seu status para acompanhar o progresso.

1. Selecione uma ameaça na lista.
2. Escolha o ícone de edição no painel de detalhes da ameaça para abrir o formulário de edição.
3. Você pode modificar os seguintes campos:
   - Status — Acompanhe o ciclo de vida da ameaça:
     - Aberto — A ameaça é reconhecida e precisa de atenção (padrão).
     - Resolvido — Você corrigiu o problema.

    - Rejeitada — Você analisou a ameaça e determinou que ela não é aplicável.
  - Severidade — ajuste o nível de gravidade se a avaliação do agente não corresponder ao seu contexto.
  - Declaração — Refine a descrição da ameaça.
  - Fonte, pré-requisitos, ação, impacto — atualize os detalhes da ameaça com base no conhecimento do seu domínio.
  - Ativos afetados — adicione ou remova ativos afetados (separados por vírgula).
  - Metas de segurança afetadas — Selecione as metas de segurança afetadas (confidencialidade, integridade, disponibilidade, autorização, Non-repudiation autenticação).
4. Selecione Salvar para aplicar as alterações.

Etapa 6: baixar um relatório

Após a conclusão da execução, você pode baixar um relatório em PDF resumindo a visão geral do sistema e todas as ameaças identificadas.

1. Na página de execução concluída, escolha Gerar relatório.
2. O PDF é baixado para o seu computador.

Etapa 7: revisar as verificações e registros de pré-voo

Se você precisar investigar como o agente chegou às conclusões ou depurar uma falha parcial:

1. Selecione a guia Preflight para ver o status das verificações de pré-voo que são executadas antes do início da análise de ameaças. Cada verificação mostra seu status (concluída, em andamento ou pendente) e uma barra de progresso indica quantas verificações foram concluídas. Você pode expandir uma verificação concluída para ver sua saída de registro detalhada.
2. Selecione a guia Registros para ver uma lista filtrável das tarefas que o agente executou durante a execução. Selecione qualquer tarefa da lista para ver sua saída de registro detalhada no painel lateral.

Próximas etapas

Depois de analisar os resultados do seu modelo de ameaça:

- Aborde primeiro as ameaças de alta gravidade com base nas recomendações do agente

- Atualize os status das ameaças à medida que você implementa as correções
- Execute um novo modelo de ameaça para verificar se suas alterações abordam as ameaças identificadas
- Ajuste suas fontes e documentos de escopo à medida que seu aplicativo evolui (consulte) the section called “Crie um modelo de ameaça”

Analise as descobertas de segurança do código em pull requests

Depois de ativar a revisão de código para pull requests para seus repositórios, o AWS Security Agent analisa automaticamente os pull requests e publica as descobertas de segurança diretamente no seu provedor de controle de origem. Isso permite que os desenvolvedores resolvam problemas de segurança em seu fluxo de trabalho normal sem sair da pull request.

> **Note**
>
> Esta página se aplica a GitHub pull requests, GitLab merge requests e pull requests do Bitbucket. A experiência é semelhante em todos os fornecedores.

Como a revisão de código funciona em pull requests

Quando você envia uma pull request (ou uma solicitação de mesclagem em GitLab) em um repositório com a revisão de código ativada, o AWS Security Agent inicia automaticamente a análise.

1. Acionador de análise de pull request - A revisão de código é acionada quando uma pull request é marcada como “Pronto para revisão” nos repositórios em que você habilitou o recurso de revisão de código. Os rascunhos de pull requests não são analisados.
2. Confirmação da análise — Quando o AWS Security Agent começa a analisar sua pull request, ele publica um comentário inicial: “O AWS Security Agent está analisando seu código...” Isso permite que você saiba que a análise começou e está em andamento.
3. Conclusão da revisão — Após a conclusão da análise, o AWS Security Agent publica uma análise em sua pull request com os resultados. Todas as descobertas de segurança são agrupadas em uma única análise para manter sua pull request organizada e minimizar as notificações.

Entendendo os resultados da revisão de código

O AWS Security Agent fornece diferentes tipos de resultados, dependendo do que ele encontra durante a análise.

Quando problemas de segurança são encontrados

Se o AWS Security Agent identificar problemas de segurança em suas alterações de código, ele publicará uma análise que inclui:

- Resumo - Uma visão geral de alto nível de todas as descobertas de segurança, descrevendo os tipos de problemas identificados e seu impacto potencial
- Descobertas individuais - As descobertas de segurança detalhadas aparecem como comentários agrupados na análise principal, com cada descoberta incluindo:
  - Descrição do problema de segurança
  - Local em seu código em que o problema foi encontrado
  - Orientação de remediação explicando como resolver o problema
  - Contexto relevante com base em suas configurações de revisão de código (violações de requisitos de segurança, vulnerabilidades comuns ou ambas)

> **Note**
>
> Os tipos de problemas de segurança analisados dependem das suas configurações de revisão de código. Se você configurou a validação dos requisitos de segurança, as descobertas farão referência aos requisitos de segurança personalizados da sua organização. Se você configurou as descobertas de vulnerabilidades de segurança, as descobertas identificarão vulnerabilidades de segurança comuns. Para obter mais informações sobre as configurações de revisão de código, consultethe section called “Habilitar revisão de código de pull request para GitHub repositórios”.

Quando nenhum problema de segurança é encontrado

Se o AWS Security Agent concluir a análise e não encontrar problemas de segurança em suas alterações de código, ele publicará um comentário: “Nenhum problema identificado”. Isso confirma que a revisão foi concluída com êxito e que suas alterações no código não acionaram nenhuma descoberta de segurança com base nas configurações de revisão de código definidas.

Respondendo às descobertas de segurança

Depois de analisar as descobertas de segurança publicadas pelo AWS Security Agent, você pode agir diretamente no seu provedor de controle de origem.

- Resolva as descobertas - atualize seu código com base na orientação de remediação fornecida nas descobertas e, em seguida, envie novos commits para a pull request. O AWS Security Agent analisará o código atualizado.
- Resolver conversas - Depois de abordar uma constatação de segurança, marque a conversa como resolvida para acompanhar seu progresso.

> **Tip**
>
> Cada descoberta inclui orientações de remediação específicas adaptadas ao problema de segurança identificado. Revise esta orientação cuidadosamente para entender o risco de segurança e como lidar com ele de forma eficaz.

Filtrando os resultados da revisão de código

Você pode personalizar a forma como o AWS Security Agent analisa seu código adicionando um `filtering.md` arquivo ao seu repositório. Esse arquivo permite reduzir os falsos positivos fornecendo contexto sobre sua base de código e excluindo arquivos ou pastas da análise.

Criando o arquivo de filtragem

Crie um arquivo nomeado `filtering.md` no `.awssecurityagent` diretório na raiz do seu repositório:

```
.awssecurityagent/filtering.md
```

O AWS Security Agent lê esse arquivo da ramificação principal do seu repositório (por exemplo, `main` ou`mainline`) ao analisar pull requests.

Estrutura do arquivo

O `filtering.md` arquivo usa a formatação Markdown padrão com seções específicas que o AWS Security Agent reconhece. O arquivo deve incluir um `Code Review` título seguido por uma ou ambas as seções a seguir: `IgnorePatterns` e `ContextHints` (sem espaço).

O exemplo a seguir mostra a estrutura completa de um `filtering.md` arquivo:

```
# filtering.md

## Code Review

### IgnorePatterns

**/*.md

/myapp/src/**/*.snap

/myapp/config/README

### ContextHints

- The backend is a trusted system and won't return non-standard protocols.
- URL is generated from server with presigned token, so no SSRF security
 vulnerabilities.
- AppSec has verified that we are allowed to use cache with an eviction policy.
```

Ignore padrões

A `IgnorePatterns` seção especifica arquivos e pastas que o AWS Security Agent deve ignorar durante a revisão do código. Use `glob patterns` para definir quais caminhos excluir da análise.

Requisitos de formato:

- Cada padrão deve estar em sua própria linha.
- Separe cada padrão com uma linha vazia entre eles. Isso garante que o arquivo seja renderizado corretamente quando visualizado em nossas ferramentas GitHub de revisão de código.
- Os padrões seguem o formato global padrão. Por exemplo, `**/*.md` corresponde a todos os arquivos markdown e `/myapp/src/**/*.snap` corresponde a todos os `.snap` arquivos dentro da `/myapp/src/` pasta na raiz.

- Nesta seção, oferecemos suporte a até 1000 padrões de ignoração.

Dicas de contexto

A `ContextHints` seção fornece contexto adicional sobre sua base de código que ajuda o AWS Security Agent a fazer avaliações mais precisas. Use dicas de contexto para explicar decisões de arquitetura, exceções de segurança ou outras informações que possam afetar a forma como as descobertas são interpretadas.

Requisitos de formato:

- Cada dica deve começar com um traço (`-`) seguido por um espaço.
- Escreva cada dica como uma única linha de texto de formato livre limitada a 500 caracteres.
- Cada dica deve descrever um contexto específico sobre sua base de código.
- Oferecemos suporte para até 20 dicas de contexto nesta seção.

As dicas de contexto são aplicadas depois que o AWS Security Agent conclui sua análise inicial, ajudando a filtrar descobertas que não se aplicam ao seu caso de uso específico.

Próximas etapas

Depois de analisar as descobertas de segurança do código:

- Atualize seu código com base na orientação de remediação
- Envie novos commits para acionar uma reanálise de suas alterações
- Ajuste as configurações de revisão de código, se necessário (consultethe section called “Habilitar revisão de código de pull request para GitHub repositórios”)
- Analise os requisitos de segurança da sua organização para entender os critérios de validação
- Considere o teste de penetração para uma validação abrangente da segurança dos aplicativos implantados

Crie uma revisão de código

Crie análises de código no aplicativo web do AWS Security Agent para escanear seus repositórios de código-fonte e fontes do S3 em busca de vulnerabilidades de segurança. As análises de código

realizam uma análise estática abrangente em toda a sua base de código, identificando problemas de segurança e fornecendo orientação para remediação.

Diferentemente da revisão de código baseada em pull request, que analisa alterações de código individuais (consultethe section called “Analise as descobertas de segurança do código em pull requests”), as análises de código sob demanda examinam todo o código-fonte para identificar vulnerabilidades de segurança e validar a conformidade com os requisitos de segurança da sua organização.

Neste procedimento, você criará uma revisão de código selecionando entradas de código-fonte, configurando permissões e executando a revisão.

Pré-requisitos

Antes de começar, verifique se você tem:

- Acesso ao aplicativo web do AWS Security Agent
- Pelo menos um GitHub repositório conectado ou bucket do S3 no seu Espaço do Agente

> **Tip**
>
> Se você já tem GitHub repositórios conectados ao seu Espaço do Agente, a revisão de código está pronta para uso — nenhuma configuração adicional é necessária. Escolha Iniciar no aplicativo web no cartão de revisão de código na sua página do Agent Space ou inicie o aplicativo web diretamente.
> Se você precisar conectar fontes adicionais ou configurar buckets do S3, consulte. the section called “Ativar revisão de código”

Acesse a página de análises de código

Navegue até a seção de revisões de código no aplicativo web.

1. Faça login no aplicativo web do AWS Security Agent.
2. Na barra lateral esquerda, escolha Revisões de código.
3. Você vê uma lista de revisões de código existentes com suas informações de origem, status da última execução e resumo das descobertas.

Crie uma revisão de código

Configure uma nova revisão de código configurando suas entradas e permissões de código-fonte.

1. Na página Revisões de código, escolha Criar revisão de código.

Configurar detalhes da revisão de código

Forneça um título e selecione o código-fonte a ser revisado.

1. No campo Título, insira um nome descritivo para sua revisão de código.

> **Tip**
>
> Use um nome que identifique o aplicativo, o repositório ou o escopo da revisão. Por exemplo, “billing-service-security-review” ou “infrastructure-code-audit”.

2. Na seção Fontes, selecione as entradas do código-fonte para essa análise. Escolha entre duas guias:

GitHub repositórios

Selecione entre os repositórios conectados ao seu Espaço do Agente.

1. Escolha a guia GitHub repositórios.
2. Na tabela Repositórios integrados, marque a caixa de seleção ao lado de cada repositório que você deseja incluir na revisão.
3. Use o campo de pesquisa para encontrar repositórios específicos por nome.

> **Note**
>
> Somente repositórios conectados ao seu Espaço do Agente por meio da configuração de revisão de código aparecem aqui. Para adicionar mais repositórios, escolha Gerenciar no Admin Console ou peça ao administrador que atualize a configuração do Agent Space.

Fontes do S3

Selecione arquivos ZIP dos buckets do S3 conectados ao seu Espaço do Agente. Seu administrador do Agent Space configura quais buckets do S3 estão disponíveis. Qualquer arquivo ZIP armazenado em um desses buckets pode ser usado como fonte para uma revisão de código.

1. Escolha a guia Fontes do S3.
2. Insira o URI do S3 de cada arquivo ZIP que você deseja incluir na revisão. Você pode adicionar até 30 fontes do S3.

> **Note**
>
> As fontes do S3 devem ser arquivos ZIP armazenados em buckets do S3 conectados ao seu Espaço do Agente. Para disponibilizar compartimentos adicionais, consultethe section called "Ativar revisão de código".

Configurar permissões do

Selecione a função de serviço do IAM e o grupo de CloudWatch registros opcional para essa análise de código.

1. Na seção Permissões, localize o menu suspenso Função de serviço.
2. Selecione a função do IAM nas funções de serviço configuradas.

> **Note**
>
> A função de serviço deve ter permissões para acessar seu código-fonte no S3 e gravar nos CloudWatch registros e em quaisquer outros recursos da AWS necessários para a revisão do código. As funções de serviço são configuradas durante a configuração da revisão de código no AWS Management Console.

3. (Opcional) No menu suspenso do grupo de CloudWatch registros, selecione um grupo de registros para armazenar os registros de execução da revisão de código.

> **Note**
>
> Se você não selecionar um grupo de registros, o AWS Security Agent cria um grupo de registros padrão para armazenar registros de revisão de código.

Configurar a correção automática de código

Habilite a remediação automática para que o AWS Security Agent gere correções de código para todas as descobertas assim que a análise for concluída.

1. Na seção Remediação automática de código, marque a caixa de seleção Ativar correção automática de código.

A forma como o AWS Security Agent entrega a correção depende da fonte:

- GitHub Repositórios privados — O AWS Security Agent envia uma pull request com a correção para o repositório.
- GitHub Repositórios públicos — Para evitar a divulgação da vulnerabilidade antes que ela seja corrigida, o AWS Security Agent não abre uma pull request. Em vez disso, ele anexa uma sugestão de comparação à descoberta que você pode baixar do aplicativo web e aplicar de forma privada.
- Fontes do S3 — A correção do código não está disponível. Revise os detalhes da descoberta e aplique as correções manualmente.

> **Important**
>
> As pull requests de remediação enviadas para repositórios privados são visíveis para todos com acesso de leitura. Revise as alterações antes de mesclar. A correção automática de código só está disponível quando GitHub os repositórios são selecionados como fonte.

> **Note**
>
> Quando desativado, você ainda pode acionar manualmente a correção de código para GitHub-sourced descobertas individuais após a conclusão da análise.

Configurar a validação simulada

Ative a validação simulada para confirmar dinamicamente se as vulnerabilidades descobertas podem ser exploradas em um aplicativo em execução.

1. Na seção Validação simulada, marque a caixa de seleção Ativar validação simulada.

Quando ativado, o AWS Security Agent provisiona um ambiente simulado após a conclusão da análise estática. Ele integra seu código-fonte, inicia serviços dentro do ambiente e tenta explorar as vulnerabilidades encontradas durante a verificação. As descobertas são então atualizadas com um status de validação indicando se a vulnerabilidade foi explorada com sucesso.

> **Note**
>
> Atualmente, a validação simulada está disponível somente para aplicativos autônomos que podem ser encaixados. Quando vários repositórios são selecionados como fontes, a validação simulada não está disponível.

> **Important**
>
> A validação simulada adiciona tempo de processamento à sua execução de revisão de código. A etapa de validação provisiona um ambiente e executa tentativas de exploração. Isso normalmente leva de 1 a 3 horas, dependendo do número de descobertas e da complexidade do aplicativo.

Destinos de rede permitidos

O ambiente simulado tem acesso à rede de saída restrito a uma lista de permissões predefinida. Seu aplicativo pode alcançar os seguintes domínios durante a validação:

A tabela a seguir lista os domínios permitidos e suas finalidades.

Domínio	Finalidade
`.amazonaws.com` , `.aws.amazon.com`	Serviços da AWS
`.public.ecr.aws`	Amazon ECR Public
`.docker.com` , `.docker.io`	Docker Hub
`.github.com` , `.githubusercontent.com`	GitHub
`.gitlab.com`	GitLab
`.npmjs.com` , `.npmjs.org`	registro npm
`.pypi.org` , `.pypi.python.org` , `.pythonhosted.org`	Python Package Index
`.crates.io` , `.rustup.rs`	Pacotes Rust
`.maven.org` , `.gradle.org`	Java/Gradle pacotes
`.nuget.org`	Pacotes.NET
`.rubygems.org` , `.ruby-lang.org`	Pacotes Ruby
`.golang.org` , `.pkg.go.dev` , `.goproxy.io`	Pacotes Go
`.nodejs.org` , `.yarnpkg.com`	Node.js

Domínio	Finalidade
`.alpinelinux.org` , `.debian.org` , `.ubuntu.com` , `.centos.org` , `.fedoraproject.org`	Repositórios de distribuição Linux
`.cloudfront.net`	CloudFront distribuições
`.google.com` , `.googleapis.com`	APIs do Google
`.microsoft.com` , `.visualstudio.com`	Serviços da Microsoft
`.sourceforge.net` , `.bitbucket.org`	Hospedagem de origem

> **Note**
>
> Se seu aplicativo exigir acesso à rede para domínios que não estão nessa lista, a conexão será bloqueada pelo firewall da rede. Aplicativos que dependem de APIs ou serviços externos não listados acima podem não iniciar corretamente no ambiente simulado.

Crie a revisão do código

1. Revise sua configuração para garantir a precisão.
2. Escolha Criar revisão de código.

Você será redirecionado para a página de detalhes da revisão de código, onde poderá iniciar uma execução de revisão.

Execute uma revisão de código

Depois de criar uma revisão de código, inicie uma execução para iniciar a análise.

1. Na página de detalhes da revisão de código, escolha Iniciar revisão.

2. O AWS Security Agent começa a analisar seu código-fonte.

Você também pode iniciar uma revisão na página da lista de revisões de código escolhendo Iniciar revisão ao lado da revisão de código que você deseja executar.

Monitore uma execução de revisão de código

Acompanhe o progresso da sua revisão de código à medida que ela é executada.

Revise as fases de execução

Uma execução de revisão de código avança nas seguintes fases, exibidas como um indicador de progresso:

1. Preflight — O AWS Security Agent valida o acesso ao seu código-fonte e configura o ambiente de teste. As verificações pré-voo incluem:
   - Configuração da infraestrutura de serviços
   - Validação do acesso à origem do S3
   - Configurar ambiente de teste
2. Análise estática — O AWS Security Agent escaneia seu código-fonte em busca de vulnerabilidades de segurança e violações de requisitos.
3. Validação simulada (opcional) — Se a validação simulada estiver habilitada, o AWS Security Agent provisiona um ambiente simulado e implanta seu aplicativo. Em seguida, ele tenta explorar as vulnerabilidades descobertas durante a análise estática. Essa etapa confirma se as descobertas podem ser exploradas no tempo de execução de destino. Essa fase só aparece quando você ativa a validação simulada durante a criação da revisão de código.
4. Finalização — O AWS Security Agent compila as descobertas e gera o resumo dos resultados.

Exibir detalhes da corrida

Na página de detalhes da execução, navegue entre as guias para monitorar o progresso:

- Execução de revisão de código — Visualize o resumo da execução, incluindo ID da execução, hora de criação, status, duração, horas da tarefa, detalhamento do nível de gravidade e gráfico de tipos de risco.
- Preflight — Visualize o progresso e o status da verificação de preflight de cada etapa de validação.

- Registros de revisão de código — Visualize as tarefas que o AWS Security Agent identificou e conduziu durante a análise, com registros de tarefas detalhados para cada etapa.
- Validação simulada — Quando a validação simulada estiver ativada, visualize o status do provisionamento, as tarefas de validação de descobertas individuais e seus resultados.
- Conclusões — Visualize as descobertas de segurança após a conclusão da análise (consultethe section called “Analise os resultados de uma revisão de código”).

Histórico de execução

Cada revisão de código mantém um histórico de todas as execuções. Na página de detalhes da análise de código:

- A seção Última execução mostra a execução mais recente com sua hora de início, status, duração e ID.
- A tabela Todas as execuções lista todas as execuções anteriores com sua hora de início, status, duração, resumo das descobertas e ID.
- Escolha Monitorar execução para ver os detalhes da última execução ativa.
- Escolha o link do horário de início de qualquer corrida para ver todos os detalhes.

Próximas etapas

Depois de executar uma revisão de código:

- Analise as descobertas de segurança e suas diretrizes de remediação (consultethe section called “Analise os resultados de uma revisão de código”)
- Corrija as descobertas por meio de pull requests automatizados ou correções manuais (consultethe section called “Corrija as descobertas da revisão de código”)
- Execute análises adicionais após implementar as correções para verificar a remediação
- Ajuste sua configuração de revisão de código ou fontes à medida que sua base de código evolui

Analise os resultados de uma revisão de código

Depois que uma execução de revisão de código for concluída, revise o resumo da execução e as descobertas de segurança para entender as vulnerabilidades em seu código-fonte. Cada descoberta

contém uma descrição, classificação de gravidade, localizações de códigos, raciocínio de risco e correções sugeridas para ajudar você a priorizar e resolver problemas de segurança.

Pré-requisitos

Antes de começar, verifique se você tem:

- Uma execução de revisão de código concluída ou em andamento
- Acesso ao aplicativo web do AWS Security Agent

Etapa 1: acessar a execução de revisão de código

Navegue até sua execução de revisão de código concluída para ver o resumo e as descobertas.

1. Faça login no aplicativo web do AWS Security Agent.
2. Na barra lateral esquerda, escolha Revisões de código.
3. Selecione a revisão de código que você deseja examinar.
4. Na tabela Todas as execuções, selecione a execução concluída clicando no link da hora de início. Como alternativa, escolha Monitorar execução para ver a execução mais recente.

Etapa 2: Monitorar o progresso da execução

Acompanhe o progresso da sua execução de revisão de código usando o indicador de etapas.

1. Localize o indicador de etapa horizontal abaixo do cabeçalho da página.
2. Analise o status de cada fase:
   - Preflight — Valida o acesso ao seu código-fonte e configura o ambiente de digitalização. As verificações incluem a configuração da infraestrutura de serviços, a validação do acesso à origem do S3 e a configuração do ambiente de teste, que inclui verificações de GitHub acesso.
   - Análise estática — escaneia seu código-fonte em busca de vulnerabilidades de segurança e violações de requisitos.
   - Validação simulada (opcional) — Quando ativada, provisiona um ambiente simulado e tenta explorar as vulnerabilidades descobertas no aplicativo em execução.
   - Finalizando — Compila as descobertas e gera o resumo dos resultados.

> **Note**
>
> Cada etapa exibe um indicador de status (Concluído ou Em andamento). A execução será concluída quando todas as fases mostrarem Concluído. A etapa de validação simulada só aparece quando você ativa a validação simulada durante a criação da revisão de código.

Etapa 3: revisar o resumo da execução

Navegue até a guia Execução da revisão de código para ver os resultados de alto nível.

1. Selecione a guia Execução de revisão de código.
2. A seção Resumo da execução fornece o status da execução, a duração e outros detalhes de alto nível. Ele também fornece um painel de resultados de segurança categorizados por nível de gravidade e tipos de risco.
3. A visão geral do aplicativo pelo AWS Security Agent fornece um resumo da verificação de revisão de código quando a execução é concluída

Etapa 4: Analise os resultados do preflight

Navegue até a guia Preflight para verificar se todas as verificações de acesso à origem foram aprovadas.

1. Selecione a guia Preflight.
2. Revise o indicador de progresso do Preflight mostrando o número de verificações concluídas.
3. Verifique se cada verificação mostra um status de sucesso:
   - Configuração da infraestrutura de serviços — confirma que o ambiente de teste está pronto
   - Validação do acesso à origem do S3 — confirma o acesso ao código-fonte do S3 (se aplicável)
   - Configuração do ambiente de teste — Confirma que o ambiente de análise está configurado

> **Note**
>
> Se alguma verificação de pré-voo falhar, a execução não prosseguirá para a análise estática. Revise os detalhes da verificação para identificar e resolver problemas de acesso e, em seguida, inicie uma nova execução.

Etapa 5: revisar os registros de revisão de código

Navegue até a guia Registros de revisão de código para ver as tarefas que o AWS Security Agent executou durante a análise.

1. Selecione a guia Registros de revisão de código.
2. Navegue pela lista de tarefas no painel esquerdo.
3. Selecione uma tarefa para ver seu registro detalhado no painel direito.

> **Tip**
>
> Use o campo de pesquisa para filtrar tarefas por nome. Os registros fornecem informações sobre o que o AWS Security Agent examinou e como ele chegou às suas conclusões.

Etapa 6: Navegar até as descobertas

Selecione a guia Descobertas para ver todas as descobertas de segurança da execução.

> **Note**
>
> Filtro de confiança padrão
>
> Por padrão, você vê somente as descobertas com alta confiança do agente. Para também mostrar descobertas com confiança média ou baixa do agente e falsos positivos, desative a opção Ocultar descobertas não verificadas.

1. Selecione a guia Descobertas.
2. As descobertas são exibidas em uma visualização dividida com a lista de descobertas à esquerda e os detalhes da descoberta selecionada à direita.
3. Revise as informações exibidas em cada cartão de descoberta:
   - Procurando um título — Um nome descritivo que resume o problema de segurança
   - Crachá de severidade — indicador de Color-coded severidade:
     - Crítico (vermelho) — requer ação imediata; a exploração pode levar ao comprometimento do sistema

    - Alto (vermelho) — requer atenção imediata; a exploração pode resultar em um impacto significativo na segurança
    - Médio (laranja) — Deve ser tratado em um prazo razoável; contribui para o risco geral de segurança
    - Baixo (amarelo) — Pode ser tratado como parte da manutenção regular; risco imediato mínimo
- Última atualização — Data e hora de quando a descoberta foi atualizada pela última vez

Etapa 7: revisar os detalhes da descoberta

Selecione descobertas individuais para ver informações abrangentes sobre cada vulnerabilidade.

1. Selecione uma descoberta no painel esquerdo para exibir seus detalhes no painel direito.
2. Analise as ações disponíveis:
    - Resolver descoberta — Marque a descoberta como resolvida depois de resolvê-la
    - Corrija o código — gere uma pull request com uma correção (disponível para GitHub fontes)
3. Forneça feedback usando Essa descoberta foi precisa? — Selecione Sim ou Não para ajudar a melhorar a precisão da análise futura.
4. Analise os principais atributos na seção Visão geral:
    - Confiança do agente — O nível de confiança que o AWS Security Agent tem nessa descoberta
    - Gravidade — O nível de risco com um selo codificado por cores
    - Tipo de risco — A categoria de risco de segurança
    - Status de validação — Quando a validação simulada está ativada, indica se a descoberta foi confirmada como explorável em um ambiente em execução:
        - Confirmado — A vulnerabilidade foi explorada com sucesso no ambiente simulado
        - Não reproduzida — A vulnerabilidade não pôde ser explorada no tempo de execução de destino
        - Falha na validação — A tentativa de validação foi inconclusiva devido a um erro
        - Não validado — A validação simulada não foi realizada para essa descoberta. Isso ocorre quando a validação não está habilitada ou a etapa de validação atinge o tempo limite antes de chegar a essa descoberta.
    - Resolvido — Se a descoberta foi resolvida (Yes/No)
    - Última atualização — Registro de data e hora da atualização mais recente

5. Expanda a seção Descrição para ler:
   - Uma explicação detalhada do problema de segurança
   - Como a vulnerabilidade pode ser explorada
   - O impacto potencial em seu aplicativo
   - Etapas de verificação que o agente executou para confirmar a descoberta
6. Expanda a seção Localizações do código para ver:
   - Caminhos de arquivo específicos em que o problema foi identificado
   - Uma breve descrição do que foi encontrado em cada local
   - Crachás de número de linha vinculados ao código relevante
7. Expanda a seção Evidências para revisar:
   - O código, a configuração ou as observações específicas que apoiam a descoberta
   - Uma breve explicação do motivo pelo qual cada item demonstra a vulnerabilidade
   - Os arquivos afetados e os números de linha de cada evidência
8. Expanda a seção Correção sugerida para ver:
   - As mudanças que o AWS Security Agent recomenda para remediar a descoberta
   - Exemplo de código ou configuração para cada alteração recomendada

> **Tip**
>
> Use as localizações do código para navegar rapidamente até os arquivos afetados no seu repositório. Cada local inclui contexto suficiente para entender o problema sem ler o arquivo inteiro.

Etapa 8: priorizar e abordar as descobertas

Valide e tome medidas com base nas descobertas para corrigir vulnerabilidades e melhorar a postura de segurança do seu aplicativo.

Para descobertas de alta severidade com alta confiança do agente:

1. Revise cuidadosamente as seções Descrição e Localizações de códigos.
2. Use o código de correção para gerar uma pull request com uma correção ou revise os PRs de remediação automática se você habilitou essa opção.

3. Planeje uma execução de revisão de código de acompanhamento para verificar se a correção é efetiva.

Para descobertas de severidade média com alta confiança do agente:

1. Priorize com base em sua tolerância ao risco e contexto de negócios.
2. Inclua tarefas de remediação em seu planejamento regular do sprint de desenvolvimento.
3. Considere se várias descobertas de severidade média juntas criam maior risco.

Para resultados de confiança média ou baixa:

1. Revise as seções Descrição e Localizações do código para validar as suposições e condições sob as quais a vulnerabilidade ocorre.
2. Se a vulnerabilidade for válida, priorize com base na gravidade e na tolerância ao risco e considere as tarefas de remediação.

Etapa 9: Acompanhe o progresso da remediação

Use a interface de descobertas e execute novamente para rastrear quais vulnerabilidades foram solucionadas.

1. Ao implementar as correções, use Resolver descoberta para marcar as descobertas como abordadas.
2. Execute uma nova revisão de código para verificar se as correções resolvem as descobertas e não introduzem novos problemas.
3. Revise a tabela Todas as execuções na página de detalhes da revisão de código para comparar as descobertas entre as execuções ao longo do tempo.

> **Tip**
>
> Depois de mesclar as pull requests de remediação, inicie uma nova execução da mesma revisão de código para confirmar se as vulnerabilidades foram resolvidas. Compare a contagem de resultados entre as corridas para acompanhar seu progresso.

Próximas etapas

Depois de analisar suas descobertas de análise de código:

- Priorize descobertas de alta severidade com alta confiança para remediação imediata
- Use o código Remediate para gerar correções automatizadas para GitHub-sourced descobertas (consultethe section called “Corrija as descobertas da revisão de código”)
- Execute análises de código adicionais após implementar as correções para verificar a correção
- Ajuste suas fontes ou configurações de revisão de código à medida que sua base de código evolui (consulte) the section called “Ativar revisão de código”

Corrija as descobertas da revisão de código

Depois de analisar as descobertas de segurança de uma análise de código, você pode usar o AWS Security Agent para gerar correções de código para descobertas em fontes do GitHub repositório. Para repositórios privados, o AWS Security Agent abre uma pull request com a correção proposta. Para repositórios públicos, o AWS Security Agent anexa um diff disponível para download à descoberta que você pode aplicar localmente. As descobertas das fontes do S3 devem ser corrigidas manualmente.

Pré-requisitos

Antes de começar, verifique se você tem:

- Uma revisão de código completa com descobertas
- GitHub repositórios conectados como fonte para a revisão do código
- Acesso ao aplicativo web do AWS Security Agent
- Familiaridade com a arquitetura e os requisitos de segurança do seu aplicativo

Como funciona a remediação de código

Quando você aciona a correção de código para uma descoberta, o AWS Security Agent analisa a descoberta e suas localizações de código e, em seguida, gera uma correção de código. A forma como a correção é entregue depende da fonte:

- GitHub Repositórios privados — O AWS Security Agent envia uma pull request para o repositório relevante. O pull request inclui uma descrição do problema de segurança e das alterações feitas.

- GitHub Repositórios públicos — O AWS Security Agent anexa uma sugestão de comparação à descoberta para que a vulnerabilidade não seja divulgada antes de ser corrigida. Você pode baixar o diff do aplicativo web e aplicá-lo localmente com`git apply /path/to/code_remediation_changes.diff`.
- Fontes do S3 — A correção do código não está disponível. Use a descrição da descoberta, as localizações do código e o raciocínio de risco para aplicar as correções manualmente.

> **Important**
>
> As pull requests criadas pelo AWS Security Agent são visíveis para todos os usuários que têm acesso de leitura ao repositório. Revise as alterações antes da fusão para garantir que elas estejam alinhadas aos requisitos do seu aplicativo.

Remediação automática de código

Se você habilitou a remediação automática de código ao criar a revisão de código, o AWS Security Agent gera correções para todas as descobertas elegíveis assim que a análise for concluída. Você não precisa realizar nenhuma ação adicional — as pull requests aparecem em seus repositórios privados conectados e as diferenças aparecem nas descobertas de GitHub repositórios públicos, à medida que a GitHub execução é finalizada.

Remediação manual de código

Se a correção automática de código estiver desativada, você poderá acionar a correção para descobertas individuais de GitHub fontes.

1. Navegue até a guia Descobertas de uma execução de revisão de código concluída.
2. Selecione a descoberta que você deseja corrigir.
3. No painel de detalhes da descoberta, escolha Remediar código.
4. O AWS Security Agent gera uma correção de código e envia uma pull request para o repositório ou anexa um diff disponível para download à descoberta, dependendo da visibilidade do repositório.

Revise os pull requests de remediação

Depois que o AWS Security Agent enviar uma solicitação pull de remediação para um repositório privado: GitHub

1. Navegue até seu GitHub repositório.
2. Localize a pull request criada pelo AWS Security Agent.
3. Analise as alterações, incluindo:
   - A descrição que explica a descoberta e a correção de segurança
   - O código muda abordando a vulnerabilidade
   - Qualquer contexto relevante sobre a abordagem de remediação
4. Mescle a pull request se a correção for apropriada ou feche-a e implemente uma solução alternativa.

> **Tip**
>
> Depois de mesclar as pull requests de remediação, inicie uma nova execução de revisão de código para verificar se as correções resolvem as descobertas e não introduzem novos problemas de segurança.

Aplique diferenças de remediação

Para descobertas de GitHub repositórios públicos, aplique o diff disponível para download localmente.

1. No painel de detalhes da descoberta, baixe a comparação na seção Correção de código.
2. Em seu clone local do repositório, execute. `git apply /path/to/code_remediation_changes.diff`
3. Analise as alterações aplicadas, teste-as em relação ao seu aplicativo e confirme-as por meio de seu fluxo de trabalho normal de desenvolvimento.

Limitações

- A correção de código só está disponível para descobertas de fontes do GitHub repositório. As descobertas das fontes do S3 devem ser corrigidas manualmente.

- O AWS Security Agent gera correções com base em sua análise da vulnerabilidade. Revise todas as alterações antes de mesclá-las ou confirmá-las para garantir que sejam apropriadas para sua inscrição.
- Algumas descobertas complexas podem exigir intervenção manual além da correção automática.

Próximas etapas

Depois de remediar as descobertas:

- Revise e mescle pull requests em seus GitHub repositórios ou confirme as diferenças aplicadas por meio de seu fluxo de trabalho normal
- Execute uma nova revisão de código para verificar as correções e verificar os problemas restantes
- Resolva as descobertas no aplicativo da web após confirmar a correção
- Ajuste suas fontes ou configurações de revisão de código conforme necessário (consultethe section called “Ativar revisão de código”)

Execute uma varredura de código diferencial com o S3

Execute uma varredura de código diferencial (diff) para analisar somente as linhas alteradas em seu código-fonte, em vez de realizar uma verificação completa do repositório. As varreduras diferenciais são mais rápidas do que as varreduras completas e produzem descobertas direcionadas a alterações de código específicas, tornando-as ideais para validação pré-mesclagem em fluxos de trabalho de desenvolvimento.

Diferentemente da revisão de código baseada em pull request, que é acionada automaticamente por eventos terceirizados do provedor de controle de origem (consultethe section called “Analise as descobertas de segurança do código em pull requests”), as varreduras diferenciais são iniciadas programaticamente por meio da API ou do SDK do AWS Security Agent. Você carrega um arquivo diff unificado para o S3 e faz referência a ele ao iniciar um trabalho de revisão de código.

Como funcionam os escaneamentos diferenciais

Uma varredura diferencial analisa suas alterações de código no contexto completo do seu repositório. Quando você cria um recurso de revisão de código com seu código-fonte carregado no S3, a verificação diferencial usa o repositório completo como contexto enquanto concentra as descobertas especificamente nas linhas alteradas em seu diff. Isso permite que a verificação

identifique problemas de segurança decorrentes da forma como suas alterações interagem com o código existente, como fluxos de autenticação interrompidos, tratamento inseguro de dados entre módulos ou alterações que expõem vulnerabilidades existentes.

O processo de digitalização:. Você carrega um arquivo diff unificado (a saída `degit diff`) em um bucket do S3 conectado ao seu Agent Space. Você chama a `StartCodeReviewJob` API com um `diffSource` parâmetro apontando para a localização do seu diff no S3. O AWS Security Agent analisa somente o código alterado no diff para verificar as vulnerabilidades de segurança e a conformidade com os requisitos de segurança da sua organização. As descobertas têm como escopo as linhas alteradas, com classificações de severidade, localizações de códigos e orientações de remediação.

Pré-requisitos

Antes de começar, verifique se você tem:

- Um espaço do agente com a revisão de código ativada (consultethe section called “Ativar revisão de código”)
- Um recurso de revisão de código já criado para o repositório de destino, com o código-fonte completo carregado no bucket do S3 conectado ao seu Agent Space. O repositório completo fornece um contexto que permite uma análise mais profunda de como suas alterações interagem com o código existente. Consulte the section called “Crie uma revisão de código” para obter instruções sobre como criar uma revisão de código e fazer o upload do código-fonte.
- Um bucket S3 conectado ao seu Agent Space
- Permissões do IAM para fazer upload para o bucket do S3 e chamar `securityagent:StartCodeReviewJob`
- Um arquivo diff unificado gerado a partir de suas alterações de código (por exemplo, a saída `degit diff main..feature-branch`)

> **Tip**
>
> Para obter resultados mais precisos, certifique-se de que seu recurso de revisão de código tenha a versão mais recente do código-fonte completo enviada para o S3. O diff scan usa esse repositório completo como contexto para entender a arquitetura do aplicativo, os fluxos de dados e os controles de segurança existentes ao analisar as linhas alteradas.

Etapa 1: gerar um arquivo diff

Gere uma comparação unificada que represente as alterações de código que você deseja verificar.

```
git diff main..feature-branch > changes.diff
```

Como alternativa, gere uma diferença para mudanças em etapas:

```
git diff --cached > changes.diff
```

> **Tip**
>
> O arquivo diff deve estar no formato diff unificado padrão. O AWS Security Agent analisa os caminhos do arquivo e as linhas alteradas dos cabeçalhos diff para definir o escopo de sua análise.

Etapa 2: faça o upload do diff para o S3

Faça upload do arquivo diff em um bucket do S3 conectado ao seu Espaço do Agente.

```
aws s3 cp changes.diff s3://my-security-agent-bucket/diffs/changes.diff
```

> **Important**
>
> O bucket do S3 deve ser um que já esteja conectado ao seu Espaço do Agente. A função de serviço do IAM associada ao seu espaço do agente deve ter acesso de leitura a esse local. Consulte the section called “Ativar revisão de código” para obter instruções sobre como conectar buckets S3.

Etapa 3: iniciar um trabalho de revisão de código diferencial

Chame a `StartCodeReviewJob` API com o `diffSource` parâmetro para iniciar uma verificação diferencial.

Usar o AWS CLI

```
aws securityagent start-code-review-job \
```

```
  --agent-space-id "your-agent-space-id" \
  --code-review-id "your-code-review-id" \
  --diff-source '{"s3Uri": "s3://my-security-agent-bucket/diffs/changes.diff"}'
```

Usando o AWS SDK (Python)

```
import boto3

client = boto3.client('securityagent')

response = client.start_code_review_job(
    agentSpaceId='your-agent-space-id',
    codeReviewId='your-code-review-id',
    diffSource={
        's3Uri': 's3://my-security-agent-bucket/diffs/changes.diff'
    }
)

print(f"Job started: {response['codeReviewJobId']}")
```

Usando o AWS SDK () JavaScript/TypeScript

```
import { SecurityAgentClient, StartCodeReviewJobCommand } from '@aws-sdk/client-
securityagent';

const client = new SecurityAgentClient({ region: 'us-east-1' });

const response = await client.send(new StartCodeReviewJobCommand({
  agentSpaceId: 'your-agent-space-id',
  codeReviewId: 'your-code-review-id',
  diffSource: {
    s3Uri: 's3://my-security-agent-bucket/diffs/changes.diff',
  },
}));

console.log(`Job started: ${response.codeReviewJobId}`);
```

A API retorna imediatamente com um `codeReviewJobId` e `status` de`IN_PROGRESS`.

Etapa 4: monitorar a digitalização

Pesquise o status do trabalho de revisão de código até que ele seja concluído.

```
aws securityagent get-code-review-job \
  --agent-space-id "your-agent-space-id" \
  --code-review-id "your-code-review-id" \
  --code-review-job-id "the-job-id"
```

O trabalho avança nas mesmas fases de uma revisão completa do código: Preflight → Análise estática → Finalização. As varreduras diferenciais geralmente são concluídas mais rapidamente do que as varreduras completas porque analisam menos linhas de código.

Etapa 5: Analisar as descobertas

Depois que o trabalho for concluído, liste as descobertas do trabalho de revisão de código.

```
aws securityagent list-findings \
  --agent-space-id "your-agent-space-id" \
  --code-review-id "your-code-review-id" \
  --code-review-job-id "the-job-id"
```

Cada descoberta inclui:

- Uma descrição do problema de segurança
- O nível de severidade (crítico, alto, médio ou baixo)
- Localizações de código referenciando linhas específicas no diff
- Raciocínio de risco explicando o impacto potencial
- Orientação de remediação

Para obter mais informações sobre como entender e agir de acordo com as descobertas, consultethe section called “Analise os resultados de uma revisão de código”.

Cotas e limites

- Número máximo de arquivos alterados em um diff: 3.000 arquivos ou 5 MB de tamanho total de diff
- As descobertas são limitadas a 30 por varredura, priorizadas por gravidade (Crítica > Alta > Média > Baixa)

Próximas etapas

Depois de executar uma verificação diferencial:

- Analise as descobertas e aplique a orientação de remediação (consultethe section called “Analise os resultados de uma revisão de código”)
- Ative os comentários do pull request para a revisão automática do código em cada pull request (consultethe section called “Habilitar revisão de código de pull request para GitHub repositórios”)
- Faça uma revisão completa do código periodicamente para detectar problemas fora das alterações individuais (consultethe section called “Crie uma revisão de código”)
- Use a integração do IDE para executar varreduras diferenciais diretamente do seu ambiente de desenvolvimento (consulte) the section called “Execute escaneamentos de segurança de código a partir do seu IDE”

Execute escaneamentos de segurança de código a partir do seu IDE

Execute escaneamentos de segurança de código do AWS Security Agent diretamente do seu IDE usando Kiro ou Claude Code. A integração do IDE permite que você escaneie seu código-fonte local em busca de vulnerabilidades de segurança, execute varreduras diferenciais somente no código alterado e realize análises de modelos de ameaças em documentos de design — tudo isso sem sair do ambiente de desenvolvimento. As descobertas aparecem junto com seu código com orientações de remediação, e você pode aplicar correções automatizadas diretamente do IDE.

Como funciona a integração com o IDE

A integração do IDE usa o servidor MCP (Model Context Protocol) do AWS Security Agent para conectar seu IDE ao serviço AWS Security Agent:

1. O servidor MCP empacota seu código-fonte em um arquivo ZIP.
2. O arquivo é carregado em um bucket do S3 que o servidor provisiona automaticamente em sua conta da AWS.
3. O servidor chama a API do AWS Security Agent para iniciar uma verificação.
4. O AWS Security Agent analisa o código em busca de vulnerabilidades de segurança e conformidade com os requisitos de segurança da sua organização.

5. As descobertas são devolvidas ao IDE com localizações de código, descrições, classificações de severidade e sugestões de remediação.

O servidor MCP gerencia toda a orquestração — provisionamento do espaço do agente, criação de bucket do S3, configuração da função do IAM, empacotamento de código e pesquisa de varredura — então você só precisa de credenciais da AWS configuradas localmente.

> **Note**
>
> O único pré-requisito são as credenciais da AWS configuradas em seu ambiente local (por exemplo, por meio do `aws configure` AWS SSO ou variáveis de ambiente). O servidor MCP provisiona automaticamente o espaço do agente, a função de serviço do IAM e o bucket do S3 no primeiro uso.

Pré-requisitos

Antes de começar, verifique se você tem:

- Credenciais da AWS configuradas localmente com as seguintes permissões:
  - `iam:CreateRole`, `iam:PutRolePolicy` (para configuração única da função de serviço)
  - `s3:CreateBucket`, `s3:PutObject`, `s3:PutPublicAccessBlock`, `s3:PutLifecycleConfiguration`
  - `securityagent:CreateAgentSpace`, `securityagent:UpdateAgentSpace`, `securityagent:ListAgentSpaces`, `securityagent:BatchGetAgentSpaces`
  - `securityagent:CreateCodeReview`, `securityagent:StartCodeReviewJob`, `securityagent:BatchGetCodeReviewJobs`
  - `securityagent:ListFindings`, `securityagent:BatchGetFindings`
  - `securityagent:StartCodeRemediation`(para correções automatizadas)
  - `sts:GetCallerIdentity`
- uv instalado (executador de pacotes Python)
- Python 3.10 ou posterior
- Um dos seguintes IDEs:
  - Kiro (instale o AWS Security Agent Power)
  - Claude Code (instale o plug-in do AWS Security Agent)

Instale o servidor MCP

Kiro

Instale o AWS Security Agent Power do Kiro Marketplace. O Power agrupa a configuração do servidor MCP, as instruções de direção e os ganchos do ciclo de vida para sugestões automáticas de verificação de segurança.

Como alternativa, configure o servidor MCP manualmente no seu projeto: `.kiro/mcp.json`

```
{
  "mcpServers": {
    "security-agent": {
      "command": "uvx",
      "args": ["awslabs.security-agent-mcp-server@latest"],
      "env": {
        "AWS_PROFILE": "default",
        "AWS_REGION": "us-east-1",
        "FASTMCP_LOG_LEVEL": "ERROR"
      }
    }
  }
}
```

Claude Code

Instale o plug-in AWS Security Agent, que fornece habilidades para configuração, varreduras completas, análises de diferenças, análises de modelos de ameaças e remediação.

Configure o servidor MCP no seu projeto: `.mcp.json`

```
{
  "mcpServers": {
    "awslabs.security-agent-mcp-server": {
      "command": "uvx",
      "args": ["awslabs.security-agent-mcp-server@latest"],
      "env": {
        "AWS_PROFILE": "default",
        "AWS_REGION": "us-east-1",
        "FASTMCP_LOG_LEVEL": "ERROR"
      }
```

```
    }
  }
}
```

> **Tip**
>
> Você também pode executar o servidor MCP via Docker se preferir não instalar o Python localmente. Consulte a documentação do servidor MCP para obter informações sobre a configuração do Docker.

First-time configuração

No primeiro uso, o servidor MCP provisiona automaticamente os recursos necessários da AWS:

Recurso	Convenção de nomes	Finalidade
Espaço do agente	User-chosen ou criado automaticamente	Recipiente para digitalizações, análises e testes
Perfil de serviço do IAM	`SecurityAgentScanRole`	Presumido por `securityagent.amazonaws.com` para ler o código enviado
Bucket do S3	`security-agent-scans-{account}-{region}`	Armazena o código-fonte compactado (ciclo de vida de expiração automática de 30 dias)

Para acionar a configuração explicitamente, pergunte ao seu IDE:

```
Set up the security agent
```

A configuração verifica suas credenciais da AWS, cria ou reutiliza um espaço de agente, provisiona a função do IAM e o bucket do S3 e confirma a prontidão. Se você já tem Agent Spaces do AWS Management Console, a configuração os mostra e pergunta qual deles usar.

> **Note**
>
> A instalação é executada automaticamente em sua primeira verificação, se ainda não estiver configurada. Você não precisa executá-lo separadamente.

Execute uma verificação de segurança completa

Examine todo o seu projeto em busca de vulnerabilidades de segurança:

```
Scan this project for security issues
```

O IDE:

1. Arquiva seu código-fonte (excluindo `.git` artefatos de construção e outros diretórios não essenciais) `node_modules`
2. Carrega o arquivo para o S3
3. Inicia um trabalho completo de revisão de código
4. Pesquisas para conclusão (normalmente em torno de 1 hora)
5. Apresenta os resultados agrupados por gravidade

Progresso da verificação

As varreduras completas geralmente levam cerca de 1 hora, dependendo do tamanho da base de código. O IDE verifica o progresso a cada 5 minutos e notifica você quando os resultados estão prontos. Você pode solicitar o status a qualquer momento:

```
How's the scan going?
```

Execute uma varredura diferencial

Para obter um feedback mais rápido durante o desenvolvimento, escaneie somente o código que foi alterado desde o git ref:

```
Scan my changes against main for security issues
```

Por padrão, se você não especificar uma referência base, o escaneamento compara suas alterações não confirmadas com. HEAD Você pode especificar explicitamente uma base diferente:

```
Diff scan my uncommitted changes
```

As varreduras diferenciais carregam o contexto completo do repositório e o patch git diff e, em seguida, executam a análise focada somente nas linhas alteradas. Os resultados geralmente chegam em 5 a 15 minutos.

Para obter mais informações sobre a API de varredura de diferenças do S3, consulte. the section called “Execute uma varredura de código diferencial com o S3”

Execute uma análise do modelo de ameaças

Analise documentos de design para mudanças na postura de segurança usando a metodologia STRIDE:

```
Run a threat model review on my spec
```

O IDE identifica seus `design.md` arquivos `requirements.md` e (normalmente abaixo`.kiro/specs/`), os carrega junto com seu código-fonte e executa uma análise do modelo de ameaças. Os resultados identificam:

- Ameaças de segurança categorizadas por STRIDE (falsificação, adulteração, repúdio, divulgação de informações, negação de serviço, elevação de privilégio)
- Classificações de gravidade para cada ameaça
- Impacto em ativos específicos em sua base de código
- Recomendações para mitigação

> **Tip**
>
> Execute uma análise do modelo de ameaças antes de gerar tarefas de implementação a partir de uma especificação. Isso detecta problemas de design de segurança antes que o código seja escrito.

Revisar descobertas

Quando uma verificação é concluída, as descobertas aparecem em seu IDE agrupadas por gravidade:

```
# CRITICAL: SQL Injection in user-service.ts
   File: src/api/user-service.ts:45
   User input flows directly into SQL query without parameterization

# HIGH: Hardcoded credentials in config.ts
   File: src/config/database.ts:12
   Database password stored in source code
```

Um relatório detalhado também é escrito `.security-agent/findings-{scan_id}.md` com informações completas para cada descoberta, incluindo tipo de risco, pontuação de confiança, localizações do código e código de remediação.

Para ver as descobertas de um escaneamento anterior:

```
Show my security findings
```

Aplique correções automatizadas

Depois de analisar as descobertas, aplique a remediação automatizada:

```
Fix the security findings
```

O IDE trabalha com as descobertas de cima para baixo por gravidade (Crítico → Alto → Médio → Baixo), aplicando correções de código usando a orientação de remediação de cada descoberta. Depois que todas as correções são aplicadas, ele executa automaticamente uma verificação de comparação para confirmar se os problemas foram resolvidos.

Você também pode corrigir descobertas individuais:

```
Fix the SQL injection in user-service.ts
```

> **Important**
>
> Sempre revise as correções automatizadas antes de se comprometer. Verifique se as correções não interrompem a funcionalidade existente nem introduzem regressões.

Sugestões automáticas de escaneamento (Kiro)

Ao usar o Kiro Power, o AWS Security Agent pode sugerir automaticamente escaneamentos de diferenças após você fazer alterações de código sensíveis à segurança. O Power instala um gancho que avalia cada rodada de codificação concluída e sugere uma varredura quando as alterações afetam:

- Lógica de autenticação ou autorização
- Criptografia ou tratamento de segredos
- Validação de entrada ou higienização de dados
- Solicitações de rede ou integrações externas
- Configuração de segurança (CORS, cabeçalhos, tratamento de sessão)

Você pode optar por não receber sugestões automáticas a qualquer momento excluindo `.kiro/hooks/security-diff-scan-suggester.kiro.hook`.

Tipos de escaneamento disponíveis

Tipo de verificação	Duração	Caso de uso	Command
Verificação completa	Cerca de 1 hora	Análise abrangente de toda a base de código	“Examine meu projeto em busca de problemas de segurança”
Análise de diferenças	5 a 15 minutos	Somente código alterado (pré-confirmação, pré-PR)	“Verificar minhas alterações” ou “Comparar a verificação com a principal”

Tipo de verificação	Duração	Caso de uso	Command
Modelo de ameaça	5 a 30 minutos	Documentos de design (requirements.md, design.md)	“Execute uma análise do modelo de ameaças de acordo com minha especificação”

Variáveis de ambiente

Variável	Description	Padrão
`AWS_REGION`	Região da AWS para chamadas de SecurityAgent API	`us-east-1`
`AWS_PROFILE`	Nome do perfil de credencial da AWS	Perfil padrão
`FASTMCP_LOG_LEVEL`	Nível de log do servidor MCP (DEBUG, INFO, WARNING, ERROR)	`WARNING`

Solução de problemas

“Não configurado. Execute a configuração primeiro.”

`.security-agent/config.json`O seu está desaparecido ou o Espaço do Agente não existe mais. Peça ao IDE para executar a configuração:

```
Set up the security agent
```

A varredura falha com AccessDenied

Certifique-se de que suas credenciais da AWS tenham as permissões necessárias do IAM listadas na seção Pré-requisitos. A permissão ausente mais comum é `iam:CreateRole` (necessária somente para a primeira configuração).

O escaneamento atinge o tempo limite ou demora muito

- Escaneamentos completos em grandes bases de código podem levar mais de 1 hora
- Use varreduras diferenciais para desenvolvimento iterativo — elas são concluídas em minutos
- Verifique se seu arquivo de origem não inclui diretórios desnecessários (o servidor MCP exclui padrões comuns automaticamente)

Diferença vazia — sem alterações na digitalização

Se você executar uma varredura de comparação sem alterações não confirmadas, o servidor MCP reportará “Sem alterações versus HEAD” e não iniciará uma verificação. Confirme ou prepare suas alterações primeiro, ou especifique uma referência base diferente.

Cotas e limites

- Tamanho máximo do arquivo de origem: 2 GB
- Ciclo de vida do bucket S3: o código carregado é excluído automaticamente após 30 dias

Próximas etapas

Depois de executar sua primeira verificação de segurança do IDE:

- Ative os comentários de revisão de código do pull request para GitHub integração automatizada (consultethe section called “Habilitar revisão de código de pull request para GitHub repositórios”)
- Configure os requisitos de segurança para a validação de políticas específicas da organização (consulte) the section called “Gerencie os requisitos de segurança”
- Execute verificações completas periódicas para detectar problemas em toda a sua base de código (consulte) the section called “Crie uma revisão de código”
- Use análises de modelos de ameaças nas especificações de novos recursos antes da implementação

Crie um teste de penetração

Configure testes de penetração automatizados para seus aplicativos web configurando o escopo do teste, os domínios de destino e o acesso aos recursos da AWS. Os testes de penetração ajudam a identificar vulnerabilidades de segurança na execução de aplicativos, simulando cenários de ataque do mundo real contra seus domínios verificados.

O AWS Security Agent realiza testes de segurança abrangentes em seus aplicativos da web com base no escopo e nas permissões configurados, fornecendo descobertas detalhadas sobre vulnerabilidades exploráveis antes que os invasores possam descobri-las.

Neste procedimento, você criará um teste de penetração configurando os detalhes do teste, definindo o escopo do teste e configurando as permissões necessárias.

Pré-requisitos

Antes de começar, verifique se você tem:

- Acesso ao aplicativo web do AWS Security Agent
- Pelo menos um domínio verificado para teste
- Função do IAM com permissões apropriadas para o AWS Security Agent
- Compreensão da arquitetura e dos caminhos críticos do seu aplicativo

Comece a criar um teste de penetração

Navegue até a página de criação do teste de penetração no Agent Web App.

1. Faça login no aplicativo web do AWS Security Agent.
2. Navegue até a seção Testes de penetração.
3. Escolha Criar um teste de penetração.

> **Tip**
>
> Somente domínios verificados podem ser incluídos nos testes de penetração. Peça ao seu administrador que verifique o domínio no console de gerenciamento da AWS. Consulte the section called “Habilite um domínio de aplicativo para testes de penetração”.

Nomeie seu teste de penetração

Forneça um nome descritivo que ajude a identificar o propósito e o escopo desse teste de penetração.

1. No campo Nome do teste de penetração, insira um nome descritivo para seu teste de penetração.

   Example

   O nome deve identificar claramente o aplicativo, o ambiente ou o componente que está sendo testado. Máximo de 100 caracteres.

Configurar o escopo do teste de penetração

Defina quais domínios e caminhos de URL serão testados e configure exclusões opcionais para controlar os limites do teste.

Adicionar domínios de destino

Especifique os domínios verificados que serão testados ativamente quanto a vulnerabilidades de segurança.

1. Na seção Escopo do teste de penetração, localize URLs de destino.
2. Expanda a seção Domínios verificados para ver os domínios disponíveis.
3. No campo URL de destino, insira uma URL de domínio de destino.

   > **Important**
   >
   > Somente domínios verificados podem ser testados. O URL deve estar em um domínio que você verificou anteriormente no AWS Security Agent. Sub-domains de um domínio verificado não exigem verificação separada.

4. Para adicionar vários domínios de destino:
   a. Escolha Adicionar domínio.
   b. Insira cada URL de domínio adicional.
5. Para remover um domínio de destino, escolha Remover ao lado da URL do domínio.

> **Tip**
>
> Para obter melhores resultados, inclua todos os domínios que fazem parte do fluxo de usuários do seu aplicativo, incluindo subdomínios para APIs, serviços de autenticação e entrega de conteúdo. Sub-domains de um domínio principal verificado não exigem verificação separada.

Excluir tipos de risco (opcional)

Escolha categorias de risco específicas para excluir do teste se elas não forem aplicáveis ao seu aplicativo.

1. Localize o campo Excluir tipos de risco.
2. Escolha a lista suspensa para ver os tipos de risco disponíveis.
3. Selecione um ou mais tipos de risco a serem excluídos do teste de penetração.

> **Note**
>
> A exclusão dos tipos de risco limita o escopo dos testes. Exclua somente os tipos de risco que não sejam relevantes para seu aplicativo ou que você queira testar separadamente.

Adicione caminhos de URL fora do escopo (opcional)

Especifique caminhos de URL que não devem ser testados durante o teste de penetração. O AWS Security Agent exclui o caminho especificado e todos os caminhos aninhados abaixo dele. Por exemplo, se você adicionar `https://example.com/admin` como um URL fora do escopo, `https://example.com/admin/tools` também estará fora do escopo.

1. Localize a seção Out-of-scope URLs.
2. No campo de entrada de Out-of-scope URLs, insira um caminho de URL a ser excluído (por exemplo, `/admin/delete` ou`/api/reset`).

> **Warning**
>
> Out-of-scope os caminhos não são testados quanto a vulnerabilidades. Certifique-se de excluir somente os caminhos que não devem ser acessados durante o teste, como operações destrutivas ou funções administrativas confidenciais.

3. Para adicionar vários caminhos fora do escopo:
   a. Escolha Adicionar URL.
   b. Insira cada caminho adicional.
4. Para remover um caminho, escolha Remover ao lado do caminho.

Adicionar domínios acessíveis (opcional)

Especifique os domínios que são necessários para o teste, mas não são alvos do teste de vulnerabilidade.

1. Localize a seção URLs acessíveis.
2. No campo de entrada URLs acessíveis, insira um domínio que deve estar acessível durante o teste.

> **Note**
>
> Adicione domínios acessíveis para serviços de terceiros (como Okta, Auth0, Stripe) que estejam fora do seu domínio de destino. Isso é necessário para que o AWS Security Agent possa acessar esses URLs para login e navegação durante o teste. O AWS Security Agent NÃO testa a penetração desses domínios — eles são usados somente para fins de acesso. Os domínios acessíveis não exigem verificação de propriedade, mesmo que pertençam a um domínio diferente do seu alvo. Somente os domínios-alvo exigem propriedade verificada.

3. Para adicionar vários domínios acessíveis:
   a. Escolha Adicionar URL.
   b. Insira cada domínio adicional.
4. Para remover um domínio, escolha Remover ao lado do domínio.

Adicione cabeçalhos HTTP personalizados (opcional)

Especifique cabeçalhos HTTP personalizados que serão adicionados a qualquer solicitação feita pelo AWS Security Agent durante o teste de penetração.

1. Localize a seção Cabeçalhos HTTP personalizados.
2. No campo de entrada de cabeçalhos HTTP personalizados, insira um nome e um valor de cabeçalho que serão associados às solicitações de saída.

> **Note**
>
> Por padrão, o AWS Security Agent adiciona um cabeçalho personalizado para User-Agentdefinir como securityagent, a menos que um valor de User-Agentcabeçalho personalizado diferente seja especificado.

3. Para adicionar vários cabeçalhos personalizados:
   a. Escolha Add header (Adicionar cabeçalho).
   b. Insira cada cabeçalho personalizado adicional.
4. Para remover um cabeçalho personalizado, escolha Remover ao lado do cabeçalho.

Configurar a função do IAM

Selecione a função de serviço pré-configurada para esse teste de penetração. O AWS Security Agent usa um modelo de Space-based permissão de agente em que os administradores configuram as funções do IAM ao configurar seu espaço de agente. Você seleciona funções que já estão configuradas e prontas para uso.

1. Na seção Permissões, localize o menu suspenso Funções de serviço.
2. Selecione a função do IAM que concede ao AWS Security Agent acesso aos recursos necessários da AWS.

> **Important**
>
> A função do IAM selecionada deve ter permissões para acessar recursos de VPC, CloudWatch registros e quaisquer outros serviços da AWS necessários para o teste de

penetração. Verifique se a função tem a relação de confiança correta com o AWS Security Agent.

3. Localize o menu suspenso do grupo de CloudWatch registros.
4. Selecione o grupo de registros em que os registros do teste de penetração serão armazenados. (opcional)

> **Note**
>
> O grupo de CloudWatch registros selecionado armazenará registros detalhados da execução do teste de penetração, incluindo solicitações feitas, respostas recebidas e vulnerabilidades descobertas.
> Se você não selecionar um grupo de registros, um novo grupo de CloudWatch registros será criado automaticamente com o `/aws/securityagent` prefixo para armazenar os registros do teste de penetração.

Remediação automática de código

Marque a caixa de seleção Ativar remediação automática.

> **Important**
>
> Para corrigir as descobertas de segurança em seus repositórios de código-fonte, o AWS Security Agent pode enviar pull requests para seus repositórios. As pull requests podem estar visíveis para todos os usuários que têm acesso de leitura aos repositórios.

Configurar recursos de VPC (opcional)

Se seus domínios de destino forem privados e hospedados em uma VPC, defina as configurações da VPC onde o AWS Security Agent deve executar testes de penetração. Essa etapa só é necessária para aplicativos que não estão acessíveis ao público.

> **Note**
>
> Ignore essa etapa se seus domínios de destino estiverem acessíveis ao público. A configuração da VPC só é necessária para testar aplicativos privados hospedados em uma Amazon Virtual Private Cloud.

Escolha a VPC, as sub-redes e os grupos de segurança para o ambiente de teste de penetração.

1. Na seção VPC, localize o menu suspenso VPC ID.
2. Selecione a VPC em que seus domínios de destino estão hospedados.

> **Important**
>
> A VPC selecionada deve conter os domínios de destino que você especificou na Etapa 3. Certifique-se de que a VPC tenha roteamento e configuração de rede adequados para permitir que o AWS Security Agent acesse seus aplicativos.

3. Localize o menu suspenso Sub-redes.
4. Selecione uma ou mais sub-redes nas quais o teste de penetração deve ser executado.

> **Note**
>
> Escolha sub-redes que tenham acesso de rede aos seus aplicativos de destino. O teste de penetração será executado a partir de recursos implantados nessas sub-redes.

5. Localize o menu suspenso Grupo de segurança.
6. Selecione o grupo de segurança que controla o acesso à rede para o teste de penetração.

> **Important**
>
> O grupo de segurança selecionado deve permitir tráfego de saída para seus domínios de destino e quaisquer domínios acessíveis. Certifique-se de que as regras do grupo de segurança permitam o acesso necessário à rede para testes abrangentes.

Configurar credenciais de autenticação (opcional)

Se seus domínios de destino exigirem autenticação, forneça credenciais para permitir que o AWS Security Agent acesse áreas protegidas do seu aplicativo durante o teste de penetração. Essa etapa só é necessária para aplicativos que exigem autenticação do usuário.

> **Note**
>
> Ignore esta etapa se seus domínios de destino não precisarem de autenticação ou se todas as áreas que você deseja testar estiverem acessíveis ao público. Configure as credenciais somente quando precisar do AWS Security Agent para testar seções autenticadas do seu aplicativo.

Adicionar credenciais da

Forneça credenciais de autenticação que o AWS Security Agent usará para acessar seu aplicativo.

1. Na seção Credencial #1, selecione um método de entrada de credencial:
   - Insira suas credenciais — insira suas credenciais diretamente no AWS Security Agent.
   - Configuração avançada — Para informações confidenciais de credenciais, use opções avançadas, como o AWS Secrets Manager ou as funções do AWS Lambda. Para mais detalhes, consulte the section called “Forneça credenciais de autenticação para testes de penetração”.

     > **Tip**
     >
     > Para ambientes de produção ou credenciais confidenciais, recomendamos usar a opção de configuração avançada para referenciar com segurança as credenciais armazenadas no AWS Secrets Manager ou no Systems Manager Parameter Store.

Insira os detalhes da credencial

Forneça o nome de usuário e a senha da conta autenticada.

1. No campo Nome do usuário, insira o nome de usuário para autenticação.
2. No campo Senha, insira a senha para autenticação.

> **Important**
>
> Certifique-se de que as credenciais fornecidas tenham níveis de acesso adequados para as áreas que você deseja testar. As credenciais devem representar o nível de acesso de um usuário típico, em vez de privilégios administrativos.

Selecione o domínio de acesso

Especifique qual domínio de destino usará essas credenciais para autenticação.

1. No menu suspenso Domínio do Access, selecione o domínio em que essas credenciais serão usadas.

> **Note**
>
> Se você tiver vários domínios de destino que exigem credenciais diferentes, você pode adicionar conjuntos de credenciais adicionais clicando em Adicionar outra credencial depois de concluir essa configuração de credenciais.

Configurar o prompt de login do agente (opcional)

Forneça instruções para orientar o AWS Security Agent no processo de autenticação do seu aplicativo.

1. Expanda a seção Solicitação de login do agente se seu fluxo de autenticação exigir instruções específicas.
2. Insira as instruções descrevendo como usar as credenciais fornecidas no fluxo de login do seu aplicativo.

> **Note**
>
> O prompt de login do agente informa ao agente como aplicar suas credenciais ao seu aplicativo. Isso é útil para fluxos de autenticação complexos, processos de login em várias etapas ou aplicativos com procedimentos de login não padrão. Inclua instruções passo

a passo, como “Navegue até /login, digite o nome de usuário no campo 'E-mail', digite a senha e escolha 'Entrar'”.

Adicione várias credenciais (opcional)

Se seu aplicativo exigir vários conjuntos de credenciais ou domínios diferentes precisarem de autenticação separada, adicione outros conjuntos de credenciais.

1. Depois de concluir a primeira configuração de credencial, escolha Adicionar outra credencial.
2. Repita as etapas de configuração de credenciais para cada conjunto adicional de credenciais.
3. Para remover um conjunto de credenciais, escolha Remover ao lado do cabeçalho da credencial.

> **Tip**
>
> Configure várias credenciais ao testar diferentes funções de usuário, acessar vários domínios autenticados ou verificar controles de acesso baseados em funções em seu aplicativo.

Anexar recursos adicionais (opcional)

Forneça recursos complementares para ajudar o AWS Security Agent a realizar testes de penetração mais completos e precisos. Recursos adicionais podem incluir diagramas de arquitetura, documentação da API, arquivos de configuração, GitHub repositórios ou S3-hosted materiais que fornecem contexto sobre seu aplicativo.

> **Note**
>
> Recursos adicionais são opcionais, mas recomendados. Fornecer informações abrangentes sobre seu aplicativo ajuda a garantir uma cobertura completa dos testes, reduz os falsos positivos e fornece resultados mais acionáveis.

Adicione recursos ao teste de penetração

Selecione os recursos existentes ou faça o upload de novos arquivos que ajudarão a orientar o teste de penetração.

1. Na seção Recursos conectados, você pode:
   - Escolha Selecionar entre os disponíveis para escolher entre os recursos já conectados ao AWS Security Agent (como GitHub repositórios ou buckets S3).
   - Escolha Carregar para adicionar novos arquivos diretamente do seu sistema local.

> **Tip**
>
> Os recursos úteis incluem documentação da API, diagramas de arquitetura, OpenAPI/Swagger especificações, arquivos de configuração, diagramas de fluxo de autenticação e qualquer outro material que descreva a estrutura e o comportamento do seu aplicativo.

Selecione entre os recursos disponíveis

Escolha entre recursos que já estão integrados ao AWS Security Agent.

1. Escolha Selecionar dos recursos existentes.
2. Navegue pela lista de recursos disponíveis de fontes conectadas, como:
   - GitHub repositórios, na guia GitHub repositórios
   - Buckets do S3
   - Arquivos enviados anteriormente
   - Repositórios de documentação
3. Selecione os recursos que você deseja incluir no teste de penetração.
4. Escolha Adicionar ao teste de penetração para anexar os recursos selecionados.

Example

Recomendamos selecionar e adicionar GitHub repositórios relevantes ao seu pentest, para que o AWS Security Agent possa desenvolver uma compreensão do contexto do seu aplicativo e gerar correções de código prontas para implementação por meio de pull requests (quando ativadas)

> **Note**
>
> Os recursos selecionados das fontes disponíveis permanecem sincronizados com sua localização original. Se você atualizar um GitHub repositório ou arquivo S3, o teste de penetração usará a versão atualizada.

> **Note**
>
> Se você tiver uma VPC privada associada ao seu pentest e um GitHub repositório configurado, certifique-se de que ela GitHub esteja acessível por meio da sua VPC privada para extrair recursos. GitHub Na maioria dos casos, você precisará garantir que o tráfego de saída via VPC NAT Gateway seja permitido por padrão ou configurar regras específicas para permitir tráfego de saída GitHub para IPs (consulte Meta API Endpoint) GitHub

Carregar novos recursos

Faça upload de arquivos diretamente do seu sistema local ou forneça conteúdo em texto simples para o AWS Security Agent.

1. Escolha Carregar.
2. Escolha um dos seguintes métodos de entrada:
   - Carregar arquivos locais - Selecione um ou mais arquivos do seu sistema local.
   - Colar texto sem formatação - Digite ou cole o conteúdo do texto diretamente no campo de entrada. Escolha Carregar.
3. Em seguida, escolha Adicionar para concluir o carregamento.
4. Os recursos carregados aparecem na tabela Recursos conectados.

> **Tip**
>
> Use a opção de texto sem formatação quando quiser fornecer rapidamente listas de endpoints de API, padrões de URL, instruções de teste ou outras informações baseadas em texto sem criar um arquivo separado.

> **Important**
>
> Certifique-se de que os arquivos enviados e o conteúdo colado não contenham informações confidenciais, como credenciais de produção, chaves privadas ou informações de identificação pessoal (PII). Use versões higienizadas dos arquivos de configuração e da documentação.

Conecte os recursos existentes

Os recursos existentes podem ser do que você enviou anteriormente para o AWS Security Agent, do seu bucket do S3 e dos seus GitHub repositórios integrados. Escolha Selecionar dos recursos existentes para selecioná-los.

Gerencie recursos conectados

Revise, organize e remova os recursos associados ao teste de penetração.

A tabela Recursos conectados exibe todos os recursos incluídos no teste de penetração com as seguintes informações:

- Nome - O nome do arquivo ou identificador do recurso
- Tipo - A categoria do recurso (arquivos enviados, recursos do S3, GitHub repositórios etc.)

Para gerenciar recursos:

1. Selecione um ou mais recursos usando as caixas de seleção.
2. Escolha Remover do teste de penetração para separar os recursos selecionados.

> **Note**
>
> Você pode classificar a tabela por Nome ou Tipo clicando nos cabeçalhos das colunas. Isso ajuda a organizar os recursos ao trabalhar com muitos arquivos.

Crie o teste de penetração

Finalize e inicie sua configuração de teste de penetração.

Depois de definir todas as configurações, você está pronto para criar o teste de penetração.

1. Revise todas as seções de configuração para garantir a precisão.
2. Escolha uma das seguintes opções:
   - Escolha Criar penetração para salvar a configuração sem executá-la imediatamente.
   - Escolha Criar e executar para salvar a configuração e iniciar imediatamente o teste de penetração.
   - Escolha Cancelar para descartar a configuração do teste de penetração.

> **Important**
>
> Antes de executar um teste de penetração, verifique se:
>
> - Todos os domínios de destino estão corretamente verificados e acessíveis
> - As funções do IAM têm as permissões apropriadas
> - Out-of-scope os caminhos estão configurados adequadamente para evitar testes de operações destrutivas
> - Você tem autorização para realizar testes de segurança em todos os domínios de destino

> **Note**
>
> Após o início do teste de penetração, você pode monitorar seu progresso na seção Execuções do teste de penetração. O teste pode levar várias horas para ser concluído. A maioria é concluída em 16 horas, dependendo do escopo e da complexidade da sua aplicação.

Analise os resultados de um teste de penetração

Monitore a execução do pentest em tempo real na página de registros do teste de penetração depois que o AWS Security Agent iniciar um pentest. O AWS Security Agent registra todas as ações durante o pentest. Após a conclusão, revise o resumo do pentest, que inclui visão geral do aplicativo, cobertura com endpoints identificados e avaliação de risco das descobertas de segurança.

Avalie as descobertas de segurança para resolver as vulnerabilidades do aplicativo. Cada descoberta contém avaliação de impacto, classificação de gravidade, evidências de apoio e detalhes da pull request de remediação (quando a remediação automática de código está ativada).

Pré-requisitos

Antes de começar, verifique se você tem:

- Um teste de penetração concluído ou em andamento
- Acesso ao aplicativo web do AWS Security Agent

Etapa 1: acessar a execução do teste de penetração

Navegue até o teste de penetração para ver a visão geral, os registros e as páginas de descobertas.

1. Faça login no aplicativo web do AWS Security Agent.
2. Navegue até a seção Testes de penetração.
3. Selecione o teste de penetração que você deseja examinar na lista.

> **Tip**
>
> A página de detalhes do teste de penetração exibe um resumo do status do teste, da data de conclusão e do número de descobertas identificadas.

Etapa 2: monitorar o progresso do teste

Acompanhe o progresso do seu teste de penetração usando o indicador de etapas.

1. Localize o indicador de etapa horizontal abaixo do cabeçalho da página.
2. Analise o status de cada fase de teste:
   - Preflight — Configuração inicial e verificações de conectividade
   - Análise estática — Análise de código e configuração
   - Pentests — Teste de tempo de execução e verificação de vulnerabilidades
   - Finalização — Validação final e geração de relatórios

> **Note**
>
> Cada etapa exibe um indicador de status (Concluído, Em andamento ou Pendente). As descobertas são descobertas e validadas durante todo o processo de teste, com novas vulnerabilidades aparecendo à medida que cada fase é concluída.

Etapa 3: Navegue até a guia de visão geral da execução do teste de penetração

1. A seção Resumo da Execução fornece o status do teste, a duração e outros detalhes de alto nível. Ele também fornece um painel de resultados de segurança categorizados por nível de gravidade e tipos de risco
2. A visão geral do aplicativo pelo AWS Security Agent fornece um resumo do teste de penetração executado
3. Os endpoints descobertos pelo AWS Security Agent fornecem uma lista de todos os endpoints descobertos e testados pelo agente de segurança da AWS durante a execução do pentest

Etapa 4: Navegue até a guia de registros do teste de penetração

Acesse registros detalhados de todas as ações que o AWS Security Agent executou durante o pentest.

1. As ações são categorizadas por tipo de ação e tipos de risco.
2. Selecione uma ação específica para ver registros detalhados:
   - Resumo do teste — High-level resumo das ações e resultados do agente
   - Registros de testes de penetração — registros detalhados de todas as atividades de teste

> **Note**
>
> As ações do validador fornecem registros que validam as descobertas em cada categoria

> **Note**
>
> A guia Descobertas exibe uma visualização dividida com a lista de descobertas à esquerda e os detalhes selecionados da descoberta à direita.

Etapa 5: Navegue até as descobertas

Cada descoberta na lista exibe informações importantes para ajudá-lo a avaliar rapidamente sua importância.

> **Note**
>
> Filtro de confiança padrão
>
> Por padrão, você vê somente as descobertas com alta confiança do agente. Para também mostrar descobertas com confiança média ou baixa do agente e falsos positivos, desative a opção Ocultar descobertas não verificadas.

Revise as informações exibidas em cada cartão de descoberta:

- Localizando o nome — O título e o identificador da vulnerabilidade
- Crachá de confiança — indica o nível de confiança do agente na descoberta (alto, médio ou baixo)
- Emblema de gravidade — Mostra o nível de risco com código de cores:
  - Crítico (vermelho) — requer ação imediata; a exploração pode levar ao comprometimento do sistema
  - Alto (vermelho) — requer atenção imediata; a exploração pode resultar em um impacto significativo na segurança
  - Médio (laranja) — Deve ser tratado em um prazo razoável; contribui para o risco geral de segurança
  - Baixo (amarelo) — Pode ser tratado como parte da manutenção regular; risco imediato mínimo
  - Informativo (azul) — Para fins informativos; risco mínimo ou nenhum risco imediato
- Timestamp da última atualização — Mostra quando a descoberta foi modificada ou validada pela última vez
- Pré-visualização da descrição — Breve resumo da vulnerabilidade

> **Important**
>
> Priorize as descobertas com emblemas de severidade crítica ou alta e altos níveis de confiança, pois eles representam vulnerabilidades validadas que exigem remediação imediata.

Etapa 6: revisar os detalhes da descoberta

Selecione descobertas individuais para ver informações abrangentes sobre cada vulnerabilidade.

1. Selecione um nome de descoberta no painel esquerdo para exibir seus detalhes no painel direito.
2. Revise o status da validação:

   > **Note**
   >
   > Se uma descoberta exibir a mensagem “Esta descoberta ainda não foi validada pelo AWS Security Agent”, significa que a detecção da vulnerabilidade ainda está sendo confirmada. Essas descobertas podem exigir verificação manual.

3. Analise os principais atributos exibidos na parte superior:
   - Confiança do agente — O nível de confiança que o AWS Security Agent tem nessa descoberta
   - Gravidade — O nível de risco com um selo codificado por cores
   - Localizando registros — Escolha “Rastrear ações e registros” para ver registros detalhados de execução e evidências
   - Tipo de risco — A categoria ou o tipo de risco de segurança (por exemplo, desvio de autenticação, injeção de SQL)
4. Expanda a seção Descrição para ler:
   - Uma explicação detalhada da vulnerabilidade
   - Como a vulnerabilidade funciona
   - Por que isso representa um risco de segurança
   - O impacto potencial em seu aplicativo
5. Expanda a seção Raciocínio de risco para entender o cálculo da severidade:
   - Detalhamento das métricas do CVSS (Common Vulnerability Scoring System)
   - Attack Vector (AV) — Como a vulnerabilidade pode ser explorada

- Complexidade de ataque (AC) — Quão difícil é a exploração
- Privilégios necessários (PR) — Qual nível de acesso é necessário
- Interação do usuário (UI) — Se a ação do usuário é necessária
- Escopo (S) — Se a vulnerabilidade afeta outros componentes
- Impactos de confidencialidade, integridade e disponibilidade

6. Expanda a seção Etapas para reproduzir para visualizar:
   - Etapas técnicas detalhadas para recriar a vulnerabilidade
   - Exemplos de solicitação e resposta
   - Parâmetros ou condições específicos que acionam o problema
7. A seção Script de verificação fornece uma forma executável de reproduzir a descoberta. Expanda esta seção (quando disponível) para ver:
   - Instruções — Como configurar e executar o script de verificação
   - Variáveis de ambiente — Lista as variáveis necessárias que você deve configurar antes de executar o script. Valores confidenciais são editados por motivos de segurança.
   - Script de download — Escolha baixar o script de verificação executável

> **Note**
>
> Os scripts de verificação estão disponíveis somente para vulnerabilidades confirmadas. O agente deve ter gerado e validado com sucesso um script de reprodução executável.

> **Note**
>
> Os scripts de verificação são gerados usando IA generativa. Revise o script antes da execução e execute-o somente nos sistemas que você está autorizado a testar. Para obter orientação sobre como testar AI-generated scripts com responsabilidade, consulte a Política de IA Responsável da AWS.

> **Tip**
>
> O script de verificação fornece uma forma executável de reproduzir a descoberta de forma independente. Defina as variáveis de ambiente necessárias com suas próprias

credenciais e, em seguida, execute o script no sistema de destino para confirmar a existência da vulnerabilidade.

Tip

Use o link “Rastrear ações e registros” para acessar o pacote completo de evidências, incluindo solicitações HTTP, respostas e tentativas de exploração que demonstram a vulnerabilidade.

Editar descobertas

Você pode editar qualquer descoberta para corrigir seus detalhes ou refinar a avaliação do agente. Os campos a seguir são editáveis:

- nome — O título da descoberta
- descrição — Descrição detalhada da vulnerabilidade de segurança
- status — Status atual da descoberta
- RiskType — Tipo de risco de segurança identificado
- Nível de risco — Nível de gravidade (CRÍTICO, ALTO, MÉDIO, BAIXO, INFORMATIVO)
- RiskScore — Pontuação numérica de risco
- raciocínio — Justificativa para a pontuação de risco atribuída
- AttackScript — Proof-of-concept código que demonstra a vulnerabilidade
- Nota do cliente — Nota opcional explicando sua justificativa para a edição

Para editar uma descoberta:

1. Navegue até a página de detalhes da descoberta.
2. Escolha o ícone de edição para modificar qualquer um dos campos na lista anterior.
3. Salve as alterações.

> **Note**
>
> Suas edições são salvas imediatamente e refletidas na descoberta. Se a Personalização de descobertas estiver ativada, todos os campos editáveis que você alterar serão usados para refinar as preferências do agente para futuras execuções.

Veja a versão original do agente

Quando você edita uma descoberta, um seletor de versão aparece no cabeçalho dos detalhes da descoberta. Isso permite que você visualize a avaliação original do agente:

1. Localize o seletor de versão no cabeçalho dos detalhes da descoberta.
2. Selecione Original para ver a descoberta como ela foi originalmente produzida pelo agente.
3. Selecione Último para retornar à versão atual com suas edições aplicadas.

Analise descobertas personalizadas

Quando a personalização de descobertas é ativada, o AWS Security Agent aprende com suas edições nas descobertas. O agente aplica essas preferências a descobertas semelhantes em futuras execuções de testes de penetração. Quando uma descoberta é ajustada com base em suas edições anteriores, o painel de detalhes exibe uma seção de alterações de personalização. Para obter informações sobre como ativar esse recurso, consulte Habilitar ou desabilitar a personalização de descobertas.

1. Localize a seção Alterações de personalização no painel de detalhes da descoberta.

> **Note**
>
> A seção de alterações de personalização aparece somente nas descobertas de que o AWS Security Agent está alinhado às suas edições anteriores. As descobertas que não correspondem a nenhuma preferência aprendida são mostradas exatamente como o agente as produziu, sem esta seção.

2. Analise o resumo do que mudou e por quê. O resumo lista cada atributo ajustado com seu valor original produzido pelo agente, o valor personalizado e o raciocínio. Por exemplo:

Com base em suas edições anteriores, as seguintes alterações foram feitas na descoberta originalmente produzida pelo sistema: Nível de risco alterado de MÉDIO para CRÍTICO; pontuação de risco alterada de 5,3 para 9,5; Raciocínio: severidade atualizada para refletir a exposição total do código-fonte do aplicativo por meio de mapas de origem acessíveis ao público.

1. Analise o resumo para entender como a descoberta foi ajustada antes de decidir como agir de acordo com ela.
2. Para substituir uma descoberta personalizada, edite-a novamente como faria com qualquer outra descoberta (por exemplo, altere a gravidade ou a pontuação de risco). Sua edição mais recente sempre tem precedência: o AWS Security Agent aprende com ela e aplica a preferência atualizada na próxima execução, substituindo a regra anterior.

> **Note**
>
> As alterações de personalização ajustam atributos de descoberta, como riskLevel, riskScore, nome, descrição, raciocínio e AttackScript, para refletir os padrões que o AWS Security Agent aprendeu com suas edições. Qualquer campo que você tenha editado anteriormente pode ser ajustado com base em descobertas semelhantes em futuras execuções.

Como a personalização aprende com suas edições

Quando a personalização de descobertas é ativada, o AWS Security Agent aprende com todas as edições que você faz nas descobertas e aplica ajustes semelhantes às execuções futuras. Qualquer campo que você editar (conforme listado na seção Editar descobertas anterior) será usado para refinar as preferências aprendidas pelo agente.

> **Note**
>
> Para o campo de status, somente marcar uma descoberta como FALSE_POSITIVE é tratado como uma preferência que pode ser aprendida. Mudanças em outros valores de status não acionam o aprendizado.

Na próxima execução do pentest, o agente avalia as novas descobertas de acordo com suas preferências aprendidas e aplica ajustes quando aplicável. Uma seção de alterações de personalização aparece em qualquer descoberta que foi ajustada, explicando o que foi alterado.

> **Note**
>
> A personalização só aprende com as edições feitas na última execução de pentest concluída. O recurso deve estar ativado no momento em que o pentest é iniciado para que o aprendizado ocorra. Se você editar a mesma descoberta em uma execução posterior, a edição mais recente terá precedência — o agente atualizará suas preferências para refletir sua decisão mais recente.

Ativar ou desativar a personalização de descobertas

A personalização de descobertas está ativada por padrão. Para desativá-lo ou reativá-lo:

1. Navegue até a configuração do pentest.
2. Localize o botão Personalização de descobertas.
3. Para ativar a personalização das descobertas, ative o botão. Para desativar, desligue-o.

Quando desativadas, as descobertas aparecem em seu estado bruto produzido pelo agente, sem nenhuma personalização aplicada. As preferências aprendidas anteriormente são preservadas e serão aplicadas novamente se o recurso for reativado.

Etapa 7: Interpretar as métricas do CVSS

Compreender as métricas do CVSS ajuda você a avaliar a verdadeira gravidade e priorizar os esforços de remediação.

Ao revisar a seção Raciocínio, preste atenção a essas métricas principais:

- Vetor de ataque (Network/Adjacent/Local/Physical) — Indica o quão remotamente o ataque pode ser executado
- Complexidade do ataque (Low/High) — Mostra se são necessárias condições especializadas para explorar
- Privilégios necessários (None/Low/High) — Identifica o nível de acesso de que um invasor precisa
- Interação com o usuário (None/Required) — Determina se a exploração precisa do envolvimento do usuário
- Confidentiality/Integrity/Availability Impacto (None/Low/High) — Mede o impacto na segurança do seu sistema

> **Important**
>
> As descobertas com vetor de ataque de rede, baixa complexidade e alto confidentiality/integrity impacto representam as vulnerabilidades mais perigosas que requerem atenção imediata.

Etapa 8: priorizar e abordar as descobertas

Tome medidas com base nas descobertas para corrigir vulnerabilidades e melhorar a postura de segurança do seu aplicativo.

Para descobertas críticas e de alta severidade com alta confiança:

1. Revise a descrição e as etapas para reproduzir as seções completamente.
2. Acesse os registros detalhados por meio do link “Rastrear ações e registros” para reunir evidências completas.
3. Acesse correções de código prontas para implementar por meio de um desses métodos:
   - Para remediação automática: use o link de pull request na seção de remediação
   - Para solicitações manuais: escolha “Código de remediação” na página de descobertas para solicitar uma pull request. Pré-requisitos:
   - O administrador deve habilitar a correção de código para GitHub repositórios no console do AWS Security Agent
   - Os repositórios devem ser incluídos na configuração do pentest
4. Planeje um teste de penetração de acompanhamento para verificar se a correção é eficaz.

Para resultados de severidade média e baixa:

1. Priorize com base em sua tolerância ao risco e contexto de negócios.
2. Inclua tarefas de remediação em seu planejamento regular do sprint de desenvolvimento.
3. Considere se várias descobertas de baixa gravidade juntas criam maior risco.
4. Documente todos os riscos aceitos com a justificativa apropriada.

> **Important**
>
> Não ignore os achados de baixa gravidade. Muitas vezes, várias vulnerabilidades de baixa gravidade podem ser encadeadas para criar explorações mais sérias, especialmente quando combinadas com engenharia social ou acesso físico.

Etapa 9: Acompanhe o progresso da remediação

Use a interface de descobertas para rastrear quais vulnerabilidades foram abordadas e quais exigem ações adicionais.

1. Ao trabalhar na remediação, consulte a seção Etapas para reproduzir para verificar suas correções.
2. Documente sua abordagem de remediação para cada descoberta para futuras auditorias de referência e conformidade.

> **Tip**
>
> Mantenha um registro de remediação que mapeie cada descoberta de acordo com sua resolução, incluindo as alterações no código, as atualizações de configuração ou as decisões de arquitetura tomadas para lidar com a vulnerabilidade.

Próximas etapas

Depois de analisar os resultados do seu teste de penetração:

- Priorize descobertas críticas e de alta severidade com alta confiança para remediação imediata
- Baixe, revise e execute scripts de verificação em seu ambiente de teste para testar os scripts e identificar vulnerabilidades
- Crie tíquetes de rastreamento em seu sistema de gerenciamento de problemas com links para encontrar detalhes e evidências
- Implemente correções e controles de segurança para lidar com as vulnerabilidades identificadas
- Monitore o indicador de progresso da execução do teste de penetração em busca de vulnerabilidades recém-descobertas

- Agende um teste de penetração de acompanhamento para verificar se as vulnerabilidades foram corrigidas adequadamente
- Atualize o processo de teste de segurança de aplicativos e o modelo de ameaças com base nas descobertas
- Analise as métricas do CVSS para entender a postura geral de segurança do seu aplicativo

Para obter mais informações sobre a realização de testes de penetração, consultethe section called “Crie um teste de penetração”.

Para obter mais informações sobre como entender o ciclo de vida do Security Agent, consulte. the section called “Entenda a hierarquia de recursos e o ciclo de vida”

Forneça credenciais de autenticação para testes de penetração

Forneça credenciais para permitir que o AWS Security Agent teste áreas autenticadas de seus aplicativos web. Sem credenciais, o agente só pode testar páginas e APIs acessíveis ao público.

Configurar credenciais de autenticação

1. No fluxo de trabalho de criação do teste de penetração, localize a seção Credenciais de autenticação - Opcional.
2. Na seção Credencial #1, escolha seu método de entrada de credencial:
   - Insira as credenciais - insira as credenciais diretamente. Ideal para ambientes de desenvolvimento e teste.
   - Configuração avançada - Use o gerenciamento de AWS-native credenciais. Recomendado para ambientes de produção e credenciais confidenciais.

Opções avançadas

Se você selecionar Configuração avançada, poderá escolher entre três estratégias de credenciais:

- Suposição de função do IAM — Para aplicativos que usam o AWS Cognito ou a autenticação do IAM
- AWS Secrets Manager — Para armazenamento seguro de credenciais com criptografia e rotação
- Função Lambda - Para geração dinâmica de credenciais ou fluxos de autenticação complexos

Insira as credenciais diretamente

1. Selecione Credenciais de entrada.
2. Insira o nome de usuário e a senha.
3. No menu suspenso URL de acesso, selecione o URL em que essas credenciais serão usadas. Isso deve ser selecionado na lista de endpoints de destino.
4. (Opcional) No campo 2FA - opcional, forneça um segredo TOTP para aplicativos que exigem autenticação de dois fatores. Você também pode:
   - Insira o segredo TOTP diretamente (por exemplo,`JBSWY3DPEHPK3PXP`) ou insira o `otpauth://totp/` URI completo (por exemplo,`otpauth://totp/Example:user@example.com?secret=JBSWY3DPEHPK3PXP&issuer=Example`).
   - Escolha o ícone de upload para carregar uma imagem de código QR da página de configuração do aplicativo autenticador. O código QR é escaneado localmente e o URI TOTP é extraído automaticamente.

     Quando um segredo TOTP é fornecido, o agente gera automaticamente novos códigos únicos e os insere quando uma solicitação de 2FA é detectada durante o login.
5. (Opcional) Expanda o prompt de login do Agent Space para fornecer instruções de login específicas se seu aplicativo tiver um fluxo de autenticação complexo.

> **Important**
>
> Use contas de teste com acesso de representante em vez de contas pessoais ou administrativas.

Use a configuração avançada

Quando você seleciona uma estratégia de credenciais avançada (função de Secrets Manager, Lambda ou IAM), o AWS Security Agent recupera suas credenciais diretamente do recurso configurado da AWS usando a função de serviço. Use o prompt de login do Agent Space para fornecer ao agente instruções sobre como aplicar essas credenciais ao seu aplicativo — por exemplo, qual URL de login acessar e quais campos do formulário preencher.

> **Note**
>
> Os segredos do Secrets Manager e as funções do Lambda devem estar na mesma conta da AWS que a configuração do AWS Security Agent. Cross-account as credenciais não são suportadas atualmente.

1. Selecione Configuração avançada.
2. No menu suspenso Estratégia de acesso do usuário, escolha uma das seguintes opções:

Selecione a função do IAM disponível para o agente assumir

Use essa opção para aplicativos que usam o AWS Cognito, o API Gateway com autenticação IAM ou outros sistemas de AWS-native autenticação. A função do IAM deve ter uma relação de confiança que permita que o AWS Security Agent a assuma e tenha permissões para acessar o sistema de autenticação do seu aplicativo.

Selecione a credencial estática do AWS Secrets Manager conectado

Use essa opção para recuperar credenciais com segurança do AWS Secrets Manager com criptografia, rotação e auditoria de acesso.

A função do IAM deve ter `secretsmanager:DescribeSecret` permissões `secretsmanager:GetSecretValue` e permissões.

O agente recupera o valor secreto diretamente do Secrets Manager. Use o prompt de login do Agent Space para informar ao agente como aplicar essas credenciais ao fluxo de login do seu aplicativo, por exemplo, para qual URL navegar e quais campos do formulário preencher. Você pode usar qualquer formato para armazenar seu segredo, pois o agente interpretará o formato usando as instruções fornecidas no prompt de login.

Por exemplo, se o agente enviar um formulário de username/password login emhttps://example.com/login, você pode formatar seu segredo como JSON com `password` campos `username` e. Se o aplicativo exigir TOTP-based 2FA, inclua um `totpSecret` campo com o segredo TOTP diretamente ou com um URI completo`otpauth://totp/`:

```
{
  "username": "test-user",
  "password": "secure-password-here",
```

```
  "totpSecret": "JBSWY3DPEHPK3PXP"
}
```

Em seguida, configure as instruções de autenticação:. Defina o URL de acesso como `https://example.com` (ou qualquer outro URL selecionado na lista de endpoints de destino). Digite o seguinte no prompt de login do Agent Space: “Navegue até o formulário https://example.com/login e insira o nome de usuário e a senha fornecidos”.

Como outro exemplo, se você tiver uma chave de API a ser fornecida em um cabeçalho HTTP, poderá armazená-la como texto simples:

```
"api-key-here"
```

Em seguida, configure as instruções de autenticação:. Digite o seguinte no prompt de login do Agent Space: “Defina o X-API-Key cabeçalho para a chave de API fornecida para todas as solicitações”.

> **Important**
>
> Somente TOTP-based 2FA é suportado. SMS, e-mail, notificações push, chaves de hardware e autenticação OAuth não são compatíveis.

Selecione a função Lambda disponível para recuperar credenciais dinamicamente

Use essa opção para sistemas de autenticação complexos, geração dinâmica de credenciais ou integração com provedores de identidade externos.

A função do IAM deve ter `lambda:InvokeFunction` permissões e a função deve ser concluída em 30 segundos.

Quando o teste de penetração é executado, o AWS Security Agent invoca sua função do AWS Lambda de forma síncrona e passa por um evento que identifica o teste de penetração e a credencial específica que está sendo resolvida:

```
{
  "pentest_arn": "arn:aws:securityagent:us-west-2:111122223333:pentest/pt-
abcd1234-5678-90ab-cdef-EXAMPLE11111",
  "actor_identifier": "Credential1"
}
```

- `pentest_arn`— O Amazon Resource Name (ARN) do teste de penetração para o qual a credencial está sendo resolvida. Use-o para definir o escopo, autorizar ou auditar a solicitação em sua função.
- `actor_identifier`— O nome da credencial que você atribuiu a essa credencial no console (por exemplo,`Credential1`). Use-o para retornar a credencial correta quando uma única função atende a várias credenciais.

O agente usa a saída da função diretamente como credencial. Use o prompt de login do Agent Space para informar ao agente como aplicar essas credenciais ao seu aplicativo, da mesma forma que você faria com o AWS Secrets Manager. Consulte the section called “Selecione a credencial estática do AWS Secrets Manager conectado” para ver exemplos de como formatar a saída da sua função Lambda e dos tipos de autenticação compatíveis.

Configurar várias credenciais

Para testar diferentes funções de usuário ou sistemas de autenticação:

1. Escolha Adicionar outra credencial.
2. Configure a credencial adicional usando qualquer um dos métodos de entrada.
3. Para remover uma credencial, escolha Remover na seção de credenciais.

Otimização de login

A otimização de login permite que o AWS Security Agent aprenda os padrões de navegação de execuções anteriores de pentest e os aplique às execuções subsequentes. Esses padrões incluem fluxos de login e fluxos de trabalho de autenticação em várias etapas. Isso reduz o tempo que o agente gasta redescobrindo como autenticar, resultando em escaneamentos mais rápidos e uma cobertura de testes mais consistente.

Como funciona a otimização de login

Na primeira execução do pentest, o agente descobre como navegar pelo fluxo de login do seu aplicativo usando as credenciais fornecidas por você. Ele registra as etapas de navegação bem-sucedidas e gera um arquivo de habilidades que captura o fluxo de trabalho de login aprendido.

Nas execuções subsequentes, o agente lê o arquivo de habilidades salvo e aplica diretamente o padrão de navegação aprendido, pulando a fase de descoberta. Isso significa que:

- Tempo mais rápido para testes autenticados — o agente aplica padrões de navegação comprovados imediatamente, em vez de explorar do zero
- Comportamento mais consistente — o agente segue o mesmo caminho comprovado em vez de explorar alternativas

> **Note**
>
> A otimização de login aprende durante a primeira sessão de login em uma execução. As sessões de login subsequentes na mesma execução se beneficiarão do que foi aprendido na primeira sessão. Em execuções futuras, todas as sessões de login se beneficiam da habilidade salva.

Ativar ou desativar a otimização de login

A otimização de login está ativada por padrão. Para desativá-lo ou reativá-lo:

1. Na configuração do pentest, navegue até a seção Credenciais de autenticação - opcional.
2. Localize o botão Otimização de login.
3. Para ativar a otimização de login, ative o botão. Para desativá-lo, desligue-o.

Quando desativado, o agente redescobre o fluxo de login do zero em cada execução. As habilidades de navegação aprendidas anteriormente são preservadas e serão aplicadas novamente se o recurso for reativado.

> **Tip**
>
> Se o fluxo de login do seu aplicativo mudar significativamente (por exemplo, uma página de login redesenhada ou uma nova etapa de autenticação), o agente se adaptará automaticamente atualizando sua habilidade aprendida na próxima execução. Você não precisa redefinir manualmente a otimização.

Corrija a descoberta de um teste de penetração

Ao visualizar as descobertas de um teste de penetração, você pode solicitar que o AWS Security Agent tente corrigir uma descoberta. Para GitHub repositórios privados, o AWS Security Agent abre

uma pull request com a correção proposta. Para repositórios públicos, a correção está disponível como um arquivo diff para download que você pode aplicar localmente.

Você deve habilitar a busca de remediação no AWS Management Console. (Consulte the section called “Permita que os usuários iniciem a remediação dos resultados do teste de penetração e da revisão de código”.) Os usuários podem iniciar a remediação de uma descoberta específica no aplicativo web do AWS Security Agent.

Pré-requisitos

Antes de começar, verifique se você tem:

- Um teste de penetração concluído ou em andamento
- Acesso ao aplicativo web do AWS Security Agent
- Familiaridade com a arquitetura e os requisitos de segurança do seu aplicativo

Etapa 1: ativar ou desativar a remediação automática

Você pode configurar as opções de remediação de código ao criar ou modificar um teste de penetração. Se você ativar a remediação automática, o AWS Security Agent tentará corrigir automaticamente os GitHub repositórios associados se o Agente confirmar uma descoberta durante o teste. Você também pode iniciar manualmente a correção do código. Na exibição para editar detalhes do teste de penetração, na seção Remediação automática de código, ative ou desative a correção de código.

Etapa 2: selecionar repositórios para correção de código

1. Escolha Avançar até a última etapa Recursos adicionais de aprendizado.
2. Escolha Selecionar entre os recursos.
3. Escolha GitHub repositórios.
4. Selecione os repositórios que você deseja para remediação de código.
5. Salve o teste de penetração.
6. Você pode ver os repositórios associados com sucesso na guia Recursos de aprendizagem do teste de penetração.

Etapa 3: iniciar um teste de penetração e ver os resultados

Execute o teste de penetração para detectar as descobertas. Para obter mais informações, consulte the section called “Analise os resultados de um teste de penetração”.

Etapa 4: iniciar e visualizar a correção do código

1. Navegue até a descoberta.
2. Se você habilitou a remediação automática de código, uma remediação de código será iniciada assim que o AWS Security Agent confirmar a descoberta.
3. Se você quiser iniciar manualmente uma correção de código, escolha o botão Remediar código.
4. Na seção Remediação de código da descoberta, você pode ver o status da remediação do código e os links para as pull requests. Se o GitHub repositório for público, a correção do código estará disponível como um arquivo para download em vez de uma pull request. Você pode executar `git apply /path/to/code_remediation_changes.diff` para aplicar a alteração ao seu repositório localmente.

Segurança e referência

Orientação de segurança, cotas de serviço e histórico de documentos do AWS Security Agent.

Segurança no AWS Security Agent

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de uma arquitetura de data center e rede criada para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O modelo de responsabilidade compartilhada descreve isto como segurança da nuvem e segurança na nuvem.

- Segurança da nuvem — AWS é responsável por proteger a infraestrutura que executa o AWS Security Agent na AWS nuvem. Isso inclui a infraestrutura de serviços, os modelos de IA e os agentes de testes de penetração. Third-party auditores testam e verificam regularmente a eficácia de nossa segurança como parte dos programas de AWS conformidade.
- Segurança na nuvem: sua responsabilidade inclui as seguintes áreas:
  - Gerenciando o acesso ao AWS Security Agent por meio de políticas e permissões do IAM
  - Protegendo o conteúdo que você fornece ao serviço, incluindo documentos de design, repositórios de código e URLs de aplicativos para testes de penetração
  - Configurando quais repositórios e aplicativos são monitorados
  - Analisar e agir de acordo com as descobertas de segurança fornecidas pelo serviço
  - Protegendo seus aplicativos com base na orientação de remediação fornecida
  - A confidencialidade dos dados, os requisitos da sua empresa e as leis e regulamentos aplicáveis

Essa documentação ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar o AWS Security Agent.

Considerações de segurança para o AWS Security Agent e o teste de penetração assistido por IA

O AWS Security Agent é um agente de fronteira que protege proativamente seus aplicativos durante todo o ciclo de vida de desenvolvimento em todos os seus ambientes. Ele conduz análises de segurança automatizadas personalizadas de acordo com seus requisitos, com equipes de segurança

definindo centralmente padrões que são validados automaticamente durante as análises. O Security Agent realiza testes de penetração sob demanda personalizados para seu aplicativo, descobrindo e relatando riscos de segurança verificados. Essa abordagem expande a experiência em segurança em seus aplicativos para corresponder à velocidade de desenvolvimento e, ao mesmo tempo, fornecer uma cobertura de segurança abrangente. Ao integrar a segurança do projeto à implantação, ele ajuda a evitar vulnerabilidades precocemente e em grande escala.

As equipes de segurança definem os requisitos de segurança organizacional uma vez no AWS Console: bibliotecas de autorização aprovadas, padrões de registro e políticas de acesso a dados. O AWS Security Agent aplica automaticamente esses requisitos de segurança durante todo o desenvolvimento, avaliando documentos e códigos arquitetônicos de acordo com seus padrões e fornecendo orientação específica ao detectar violações. Isso proporciona uma fiscalização de segurança consistente em todas as equipes e dimensiona as análises de acordo com a velocidade de desenvolvimento.

Para validação da implantação, o AWS Security Agent transforma o teste de penetração de um gargalo periódico em um recurso sob demanda. As equipes de segurança fornecem URLs de destino, detalhes de autenticação, código-fonte e documentação. O AWS Security Agent desenvolve um profundo entendimento de aplicativos e executa cadeias de ataque sofisticadas para descobrir e validar vulnerabilidades, permitindo que as equipes testem sempre que necessário.

Capacidades gerais

O AWS Security Agent fornece recursos de segurança abrangentes que abrangem todo o ciclo de vida do desenvolvimento.

Análise de segurança do design

O AWS Security Agent fornece feedback de segurança sob demanda sobre documentos de design e avalia a conformidade com os requisitos de segurança organizacional antes que o código seja escrito. As equipes de segurança carregam documentos de design por meio do aplicativo web, onde o agente os analisa de acordo com seus requisitos de segurança e apresenta as descobertas com orientações de remediação. Isso transforma as revisões manuais de horas em análises focadas, permitindo que as equipes abordem as questões de segurança quando a remediação é mais eficiente.

Análise de segurança do código

O AWS Security Agent analisa pull requests ou código carregado em busca de requisitos de segurança organizacional e problemas de segurança comuns, como falta de validação de entrada

e riscos de injeção de SQL. O agente fornece orientação de remediação diretamente na sua plataforma de repositório de código. As equipes de segurança configuram quais repositórios monitorar, escalando a avaliação em todas as bases de código e, ao mesmo tempo, supervisionando os problemas críticos.

On-demand teste de penetração

O AWS Security Agent fornece testes de penetração sob demanda que descobrem e relatam vulnerabilidades de segurança validadas por meio de cenários de ataque personalizados em várias etapas. O AWS Security Agent implanta agentes de IA especializados que desenvolvem o contexto do aplicativo a partir da documentação e das credenciais fornecidas e, em seguida, executam cadeias de ataque sofisticadas para identificar vulnerabilidades complexas que as ferramentas convencionais ignoram. Ele documenta as descobertas com análise de impacto, caminhos de ataque reproduzíveis e correções de código prontas para implementar, acelerando os testes de penetração de semanas para horas e escalando a validação em todo o seu portfólio de aplicativos.

Perguntas frequentes

&Controle de segurança

Como o AWS Security Agent autentica e mantém o acesso aos sistemas?

O teste de penetração é o único recurso do AWS Security Agent que pode ser autenticado no sistema de um usuário em tempo de execução. O AWS Security Agent aceita credenciais na forma de credenciais estáticas de nome de usuário e senha (armazenadas no Secrets Manager) ou um fornecedor de credenciais (como uma função Lambda) como configuração antes de iniciar o teste de caneta. Essas credenciais são usadas para exercer a funcionalidade normal do usuário system/application durante o ciclo de vida do teste de caneta. Incentivamos os usuários a criar novas credenciais com permissões com escopo adequado para fins de teste.

Os usuários podem controlar o escopo e a profundidade dos testes para evitar impactos não intencionais no sistema?

O AWS Security Agent permite que os clientes selecionem uma categoria específica de vulnerabilidade para explorar em um endpoint. Os usuários podem especificar URLs fora do escopo para impedir que o AWS Security Agent realize testes de penetração contra esses alvos. https://docs.aws.amazon.com/securityagent/latest/userguide/perform-penetration-test.html

O próprio AWS Security Agent pode representar um risco de segurança?

O AWS Security Agent é instruído a descobrir riscos de segurança, mas a fazer isso usando cargas de impacto intencionalmente mínimas (como extrair a versão do SQL em vez de descartar uma tabela quando um ataque de injeção de SQL é descoberto). O AWS Security Agent também se limita a barreiras determinísticas para evitar comportamentos arriscados, como criar carga excessiva contra o aplicativo de destino. Embora existam barreiras de proteção, ainda podem haver interações lógicas de negócios não intencionais ou não óbvias, portanto, sempre recomendamos fazer testes de penetração em um ambiente de pré-produção.

Quais dados o AWS Security Agent coleta e onde eles são armazenados?

O AWS Security Agent permite que os usuários façam upload de artefatos para fornecer contexto sobre o aplicativo que está sendo testado. Para obter mais informações sobre proteção de dados, consultethe section called “Proteção de dados”. O AWS Security Agent selecionará automaticamente a região ideal em sua geografia para processar suas solicitações de inferência. Isso maximiza os recursos computacionais disponíveis, a disponibilidade do modelo e oferece a melhor experiência ao cliente. Seus dados permanecerão armazenados somente na região de origem da solicitação. No entanto, as solicitações de entrada e os resultados de saída podem ser processados fora dessa região. Todos os dados serão transmitidos criptografados pela rede segura da Amazon. Para obter mais informações, consulte Inferência entre regiões.

Quais controles estão presentes para bloquear testes não autorizados em um endpoint?

Os endpoints especificados como URLs de destino para pentesting exigirão validação de DNS ou validação de HTTP como medida de propriedade. O AWS Security Agent solicitará que o cliente adicione um registro TXT ao DNS do endpoint ou exponha uma sequência de caracteres de validação de retorno da rota HTTP como prova de propriedade. Somente após demonstrar a prova de propriedade, o usuário poderá prosseguir com um pentest. Solicitações para URLs fora dos URLs de destino e acessíveis serão bloqueadas pela rede.

Os clientes são responsáveis por garantir que tenham a autorização adequada para testar todos os sistemas que possam ser afetados por suas atividades de teste de penetração. Todo uso do AWS Security Agent deve estar em conformidade com a Política de Uso Aceitável da AWS (https://aws.amazon.com/aup/).

Como os usuários bloqueiam e denunciam qualquer abuso usando o AWS Security Agent?

O AWS Security Agent monitora continuamente solicitações e tentativas de acessar URLs que estão fora dos URLs de destino. Se for detectado abuso, como a tentativa de usar o AWS Security Agent

para realizar testes não autorizados em um endpoint de terceiros, todos os pentests em andamento na conta serão encerrados. Os clientes podem entrar em contato com o AWS Support ou com sua equipe de contas da AWS para obter ajuda.

O AWS Security Agent pode substituir o fluxo de trabalho de testes de caneta?

O AWS Security Agent não é um serviço profissional de testes de penetração, e incentivamos os usuários a integrar o AWS Security Agent em seu fluxo de trabalho de análise de segurança. O AWS Security Agent pode fornecer acessibilidade a testes de penetração sob demanda durante a fase de desenvolvimento do ciclo de vida do software, quando interagir com profissionais de pentesting seria muito cedo, impraticável ou precisaria ser reavaliado com muita frequência. Os profissionais de segurança podem analisar as descobertas do AWS Security Agent para validá-las, explicá-las ou ampliá-las para novas descobertas (se existirem).

Os usuários podem configurar o controle de acesso baseado em funções (RBAC) para diferentes membros da equipe?

Sim. O AWS Security Agent se integra ao AWS IAM Identity Center, permitindo que os administradores gerenciem membros da equipe que podem acessar o aplicativo web do AWS Security Agent, que permite aos usuários criar, gerenciar e visualizar análises de design e pentests.

Capacidades de teste

Quais tipos de vulnerabilidades o AWS Security Agent pode detectar?

O AWS Security Agent detecta vulnerabilidades no OWASP Top 10 para aplicativos web. O AWS Security Agent fornece tipos de risco específicos que você pode incluir ou excluir nos testes descritos abaixo. As descobertas podem surgir dentro dessas categorias de risco ou de novas descobertas descobertas seguindo as pistas de uma combinação dessas categorias de risco.

- Upload arbitrário de arquivos
  - O upload arbitrário de arquivos confirma que o aplicativo deve ser capaz de se defender de arquivos falsos e maliciosos de forma a manter o aplicativo e os usuários seguros
- Injeção de código
  - Injeção de código é o termo geral para tipos de ataque, que consistem em injetar código que é então enviado interpreted/executed pelo aplicativo.
- Injeção de comando
  - A injeção de comando é um ataque no qual o objetivo é a execução de comandos arbitrários no sistema operacional hospedeiro por meio de um aplicativo vulnerável

- Cross-Site Criação de scripts (XSS)
  - Cross-Site Os ataques de script (XSS) são um tipo de injeção, na qual scripts maliciosos são injetados em sites que, de outra forma, seriam benignos e confiáveis
- Referência direta de objetos inseguros
  - Referências diretas de objetos inseguras (IDOR) ocorrem quando um aplicativo fornece acesso direto a objetos com base na entrada fornecida pelo usuário
- Vulnerabilidades do JSON Web Token
  - As JWTs são uma fonte comum de vulnerabilidades, tanto na forma como são implementadas nos aplicativos quanto nas bibliotecas subjacentes
- Inclusão de arquivo local
  - A vulnerabilidade de inclusão de arquivos permite que um invasor inclua um arquivo, geralmente explorando mecanismos de “inclusão dinâmica de arquivos” implementados no aplicativo de destino
- Travessia do caminho
  - O ataque Path Traversal (também conhecido como passagem de diretório) visa acessar arquivos e diretórios armazenados fora da pasta raiz da web
- Escalonamento de privilégios
  - O escalonamento de privilégios ocorre quando um usuário obtém acesso a mais recursos ou funcionalidades do que normalmente são permitidos, e tais elevações ou alterações deveriam ter sido evitadas pelo aplicativo
- Server-Side Falsificação de solicitação (SSRF)
  - Server-Side A falsificação de solicitação (SSRF) ocorre quando o invasor pode abusar da funcionalidade do servidor para ler ou atualizar recursos internos
- Server-Side Injeção de modelo
  - As vulnerabilidades de injeção de modelo do lado do servidor (SSTI) ocorrem quando a entrada do usuário é incorporada em um modelo de maneira insegura e resulta na execução remota de código no servidor
- Injeção de SQL
  - O ataque de injeção de SQL consiste na inserção ou “injeção” de uma consulta SQL por meio dos dados de entrada do cliente para o aplicativo
- Entidade externa XML

- O ataque de entidade externa XML é um tipo de ataque contra um aplicativo que analisa a entrada XML. Esse ataque ocorre quando a entrada XML contendo uma referência a uma entidade externa é processada por um analisador XML mal configurado.

Quais métodos de autenticação são compatíveis com o AWS Security Agent?

O AWS Security Agent oferece suporte a métodos de autenticação comuns, incluindo OAuth e JWT. Para obter mais informações, consulte a documentação do .

Como o AWS Security Agent lida com a limitação de taxa e a prevenção de negação de serviço (DOS)?

O AWS Security Agent tem proteções para evitar que ele interrompa ou derrube os endpoints em teste, incluindo o DOS. Ele tem controles de velocidade internos para detectar e lidar com padrões de tráfego inesperados.

O AWS Security Agent pode testar as APIs REST e GraphQL?

Sim, o AWS Security Agent pode testar endpoints de API. Incentivamos os clientes a fornecer a documentação da API como recursos adicionais de aprendizado, permitindo que o AWS Security Agent tenha um melhor contexto sobre a forma e a funcionalidade de cada API que está sendo testada.

Como os usuários podem verificar se o AWS Security Agent cobriu toda a lógica e os endpoints críticos do aplicativo?

O AWS Security Agent fará uma exploração abrangente dos aplicativos de destino e tentará exercê-los normalmente antes de tentar qualquer exploração. Isso permite que ele construa uma compreensão prática do aplicativo em tempo de execução e descubra a lógica e os endpoints críticos do aplicativo. Devido à sua natureza estocástica, não é garantido que o AWS Security Agent descubra e teste todos os aplicativos e endpoints críticos de qualquer aplicativo de destino. O aplicativo web do AWS Security Agent fornece visibilidade de todos os endpoints descobertos e das ações realizadas nos registros do teste de penetração.

Precisão & e confiabilidade

Como o AWS Security Agent valida as descobertas antes da emissão de relatórios?

O AWS Security Agent usa validadores determinísticos para ajudar a validar a descoberta relatada. Nos tipos de risco em que não é possível usar validadores determinísticos, o AWS Security Agent

repetirá de forma independente as etapas de descoberta para ganhar confiança na validade da descoberta. O AWS Security Agent relata apenas as descobertas de alta ou média confiança e oculta as descobertas não verificadas por padrão.

O AWS Security Agent pode se adaptar à lógica personalizada do aplicativo?

O AWS Security Agent aceita opcionalmente o código-fonte, o modelo de ameaça, os documentos de design e a documentação da API como recursos adicionais de aprendizado para obter contexto direcionado ao usuário sobre o aplicativo de destino usado no ciclo de vida de um pentest.

Os usuários podem revisar a metodologia de teste do AWS Security Agent antes da execução?

Atualmente, não há como visualizar o curso de ação do AWS Security Agent. O plano do AWS Security Agent é dinâmico por natureza, com base na exploração do aplicativo de destino. Os clientes podem monitorar o AWS Security Agent à medida que ele passa por sua exploração em tempo real, observando os registros do teste de penetração. Se os registros mostrarem uma trajetória inválida ou indesejável, os clientes poderão interromper a execução contínua do pentest.

&Implantação de integração

O AWS Security Agent se integra a ferramentas de segurança (SIEM, gerenciamento de vulnerabilidades) ou CI/CD pipelines?

O AWS Security Agent não se integra a nenhuma ferramenta ou CI/CD pipeline de segurança existente.

Como o AWS Security Agent lida com configurações específicas do ambiente?

O AWS Security Agent pode ser configurado para ser executado com funções específicas do IAM, dentro de VPCs, com credenciais relevantes do aplicativo especificado pelo cliente e com repositórios de origem do Github como referência de código-fonte para o aplicativo de destino.

O AWS Security Agent pode ser executado em ambientes isolados ou isolados?

O AWS Security Agent pode ser configurado para ter conectividade com VPCs, incluindo aquelas que não têm acesso de saída à Internet.

Vários membros da equipe podem realizar testes simultaneamente?

O AWS Security Agent oferece suporte a 5 execuções simultâneas de pentest por conta, independentemente de quem inicia o teste. Os clientes podem criar no máximo 100 Agent Spaces e 1.000 projetos Pentest.

Impacto operacional

Qual é o impacto no desempenho dos sistemas testados?

O AWS Security Agent tem proteções para evitar que ele interrompa ou derrube endpoints em teste. Isso inclui controles de velocidade sobre o número de chamadas que o AWS Security Agent pode fazer para um endpoint. O sistema ou o endpoint em teste deve esperar algum aumento no tráfego e possíveis alertas de monitoramento sendo acionados devido à atividade do pen test. Nossa recomendação é executar somente o AWS Security Agent ou qualquer atividade de pen testing no ambiente de pré-produção.

Os usuários podem agendar ou limitar o AWS Security Agent?

O AWS Security Agent não tem APIs públicas nem a capacidade de programar a execução do pen test. O AWS Security Agent também não oferece um controle de simultaneidade nas solicitações ao endpoint de destino ao iniciar a execução do pen test. Se o AWS Security Agent estiver causando problemas nos endpoints de destino, os clientes podem interromper os pentestes em andamento.

Qual é a duração típica de uma avaliação de segurança completa?

O tempo de execução de cada pentest depende da amplitude do aplicativo de destino e dos tipos de risco configurados para serem avaliados. A maioria das execuções de pentest é concluída em 16 horas.

Políticas gerenciadas pela AWS para agentes de segurança da AWS

Uma política gerenciada pela AWS é uma política independente que é criada e administrada por ela. As políticas gerenciadas pela AWS são projetadas para fornecer permissões para muitos casos de uso comuns, para que você possa começar a atribuir permissões a usuários, grupos e funções.

Lembre-se de que as políticas gerenciadas pela AWS podem não conceder permissões de privilégio mínimo para seus casos de uso específicos porque estão disponíveis para uso de todos os clientes da AWS. Recomendamos que você reduza ainda mais as permissões definindo as políticas gerenciadas pelo cliente que são específicas para seus casos de uso.

Você não pode alterar as permissões definidas nas políticas gerenciadas pela AWS. Se a AWS atualizar as permissões definidas em uma política gerenciada pela AWS, a atualização afetará todas as identidades principais (usuários, grupos e funções) às quais a política está vinculada. É mais provável que a AWS atualize uma política gerenciada pela AWS quando um novo serviço da AWS for lançado ou novas operações de API forem disponibilizadas para serviços existentes.

Para obter mais informações, consulte as políticas gerenciadas pela AWS no Guia do usuário do IAM.

Política gerenciada pela AWS: SecurityAgentWebAppAPIPolicy

Concede permissões para interagir com a API do aplicativo Web do Security Agent. Essa política permite que os usuários configurem e executem testes automatizados de penetração de segurança, gerenciem execuções de testes, visualizem descobertas de segurança e acessem recursos do Security Agent. Essa política faz referência ao tipo de recurso da instância de agente legada e às ações específicas do IAM legadas.

Detalhes das permissões

Essa política concede permissões para interagir com o serviço Security Testing Control (securityagent: *) para:

- Gerenciamento de Pentest: Crie, atualize, exclua e liste testes de penetração e seus trabalhos de execução
- Descobertas de segurança: visualize, descreva e atualize as descobertas de segurança de testes concluídos, incluindo conteúdo e metadados relacionados.
- Gerenciamento de tarefas: liste e recupere tarefas de revisão de código e revisão de documentação
- Descoberta de recursos: liste e visualize instâncias, artefatos, integrações e endpoints descobertos do agente
- Execução de testes: inicie e interrompa as execuções de pentest com recursos de monitoramento em tempo real
- Remediação de código: inicie a remediação automática de código para descobertas de segurança

Para ver a versão mais recente do documento de política JSON, consulte SecurityAgentWebAppAPIPolicyo AWS Managed Policy Reference Guide.

Política gerenciada pela AWS: AWSSecurityAgentWebAppPolicy

Concede permissões para interagir com a API do aplicativo Web do Security Agent. Essa política permite que os usuários configurem e executem testes automatizados de penetração de segurança, gerenciem execuções de testes, visualizem descobertas de segurança e acessem recursos do Security Agent.

Detalhes das permissões

Essa política concede permissões para interagir com o serviço Security Testing Control (securityagent: *) para:

- Gerenciamento de Pentest: Crie, atualize, exclua e liste testes de penetração e seus trabalhos
- Descobertas de segurança: visualize, descreva e atualize as descobertas de segurança de testes concluídos, incluindo conteúdo e metadados relacionados.
- Gerenciamento de tarefas: liste e recupere tarefas de revisão de código e revisão de design
- Descoberta de recursos: liste e visualize espaços de agentes, artefatos, integrações e endpoints descobertos
- Execução do teste: inicie e interrompa trabalhos de pentest com recursos de monitoramento em tempo real
- Remediação de código: inicie a remediação automática de código para descobertas de segurança

Para ver a versão mais recente do documento de política JSON, consulte AWSSecurityAgentWebAppPolicyo AWS Managed Policy Reference Guide.

Atualizações dos agentes de segurança da AWS nas políticas gerenciadas pela AWS

Veja detalhes sobre as atualizações das políticas gerenciadas da AWS para agentes de segurança da AWS desde que esse serviço começou a rastrear essas mudanças.

Para receber notificações de todas as alterações do arquivo de origem nesta página específica da documentação, você pode se inscrever no seguinte URL com um leitor de RSS:

Alteração	Descrição	Data
Foram adicionadas permissões à AWSSecurityAgentWebAppPolicy.	Permissões adicionadas `TargetDomain` e de `DesignReviewFeedback` recursos para os novos tipos de recursos.	31 de março de 2026
Foi adicionada uma nova política gerenciad	Foi adicionada uma política gerenciada `AWSSecuri`	9 de fevereiro de 2026

Alteração	Descrição	Data
a AWSSecurityAgentWebAppPolicy.	`tyAgentWebAppPolicy` para o novo tipo de AgentSpace recurso e alterações no nome da ação do IAM.	
Foram adicionadas permissões à SecurityAgentWebAppAPIPolicy.	Adicionado `securityagent:StartCodeRemediation` para permitir que os usuários iniciem a correção automática de código para descobertas de segurança.	20 de janeiro de 2026
Foram adicionadas permissões à SecurityAgentWebAppAPIPolicy.	Adicionado `securityagent:BatchGetSecurityTestContentMetadata` para permitir que os usuários visualizem imagens no console.	5 de dezembro de 2025
Os agentes de segurança da AWS começaram a monitorar as mudanças.	Os agentes de segurança da AWS começaram a monitorar as mudanças em suas políticas gerenciadas pela AWS.	2 de dezembro de 2025

Proteção de dados no AWS Security Agent

O modelo de responsabilidade compartilhada da AWS se aplica à proteção de dados no AWS Security Agent. Conforme descrito nesse modelo, a AWS é responsável por proteger a infraestrutura global que executa toda a Nuvem AWS. Você é responsável por manter o controle sobre o conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos serviços da AWS que você usa. Para obter informações sobre proteção de dados na Europa, consulte o Modelo de Responsabilidade Compartilhada da AWS e a postagem do blog sobre o GDPR no blog de segurança da AWS. Para fins de proteção de dados,

recomendamos que você proteja as credenciais da conta da AWS e configure usuários individuais com o AWS IAM Identity Center ou o AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com os recursos da AWS. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e o registro de atividades do usuário com a AWS CloudTrail. Para obter informações sobre o uso de CloudTrail trilhas para capturar atividades da AWS, consulte Como trabalhar com CloudTrail trilhas no Guia CloudTrail do usuário da AWS.
- Use as soluções de criptografia da AWS, juntamente com todos os controles de segurança padrão nos serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sensíveis armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-3 ao acessar a AWS por meio de uma interface de linha de comando ou de uma API, use um endpoint FIPS. Para saber mais sobre os endpoints FIPS disponíveis, consulte Federal Information Processing Standard (FIPS) 140-3.

É altamente recomendável que nunca sejam colocadas informações confidenciais ou sensíveis, como endereços de e-mail de clientes, em tags ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com o AWS Security Agent ou outros serviços da AWS usando o console, a API, a AWS CLI ou os AWS SDKs. Quaisquer dados inseridos em tags ou em campos de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, é fortemente recomendável que não sejam incluídas informações de credenciais no URL para validar a solicitação nesse servidor.

Criptografia em repouso

O AWS Security Agent criptografa todos os dados em repouso usando chaves de AWS-managed criptografia por padrão. Isso inclui:

- Documentos de design e código — Todos os documentos de design, repositórios de código e artefatos de aplicativos que você fornece para análises de segurança são criptografados usando AES-256 criptografia.

- Descobertas de segurança — Todas as descobertas de segurança, relatórios de vulnerabilidade e recomendações de remediação são criptografados em repouso.
- Dados de configuração — os requisitos de segurança, as políticas personalizadas e as configurações do serviço são criptografados.
- Registros de auditoria — Todos os registros de atividades do serviço e trilhas de auditoria são criptografados.

O AWS Security Agent usa o AWS Key Management Service (AWS KMS) para gerenciar chaves de criptografia. Opcionalmente, você pode usar uma chave gerenciada pelo cliente para criptografar seus dados, oferecendo controle total sobre as chaves de criptografia que protegem seus recursos. Para obter mais informações, consulte the section called “Chaves gerenciadas pelo cliente”.

Criptografia em trânsito

O AWS Security Agent criptografa todos os dados em trânsito usando o Transport Layer Security (TLS) 1.2 ou superior. Isso se aplica a:

- Comunicações de API — Todas as chamadas de API entre seus aplicativos e o AWS Security Agent usam HTTPS com criptografia TLS.
- Acesso ao console — O console do AWS Security Agent é acessado por HTTPS.
- Conexões de repositório — As conexões com GitHub e outros repositórios de código usam protocolos criptografados.
- Comunicações do agente — Todas as comunicações entre o serviço AWS Security Agent e os agentes de teste de penetração usam canais criptografados.

Gerenciamento de chaves

O AWS Security Agent usa o AWS Key Management Service (AWS KMS) para gerenciar chaves de criptografia. Por padrão, os dados são criptografados usando AWS-managed chaves. Opcionalmente, você pode especificar uma chave gerenciada pelo cliente ao criar recursos como Agent Spaces e integrações. Para obter mais informações, consulte the section called “Chaves gerenciadas pelo cliente”.

Privacidade do tráfego entre redes

O AWS Security Agent usa a Internet pública para se comunicar com provedores de controle de origem hospedados na nuvem (GitHub, GitLab, Bitbucket) e com o Confluence Cloud.

Para provedores auto-hospedados (GitHub Enterprise Server) GitLab Self-Managed, você pode configurar conexões privadas usando o Amazon VPC Lattice para manter todo o tráfego dentro da rede da AWS. Para obter mais informações, consulte the section called “Conecte-se ao controle de origem hospedado de forma privada”.

Na configuração padrão, o AWS Security Agent usa a Internet pública para acessar seu aplicativo para testes de penetração. Opcionalmente, você pode configurar testes de penetração para usar uma VPC para acessar seu aplicativo. Para obter mais informações, consulte the section called “Conecte o agente aos recursos privados da VPC”.

Cross-Region processamento de dados

O AWS Security Agent usa inferência entre regiões para otimizar os recursos computacionais disponíveis e a disponibilidade do modelo. Dependendo da região de origem da solicitação, podemos processar solicitações de entrada e resultados de saída em uma região diferente.

- No Leste dos EUA (Norte da Virgínia) —`us-east-1`, Oeste dos EUA (Oregon) —`us-west-2`, Ásia-Pacífico (Sydney) —`ap-southeast-2`, Ásia-Pacífico (Tóquio) —`ap-northeast-1`, Europa (Frankfurt) — e Europa (Irlanda) —`eu-central-1`, o `eu-west-1` AWS Security Agent usa inferência geográfica entre regiões. Para a maioria dos recursos, o processamento de dados permanece dentro do limite geográfico (como EUA, UE, Austrália ou Japão) de onde a solicitação foi originada. Para remediação de código, as solicitações da Austrália e do Japão são processadas na União Europeia. Para obter detalhes de roteamento específicos do recurso, consulte a tabela de inferência entre regiões.
- Na Ásia-Pacífico (Mumbai) —`ap-south-1`, Ásia-Pacífico (Cingapura) — e América do Sul (São Paulo) —`ap-southeast-1`, o `sa-east-1` AWS Security Agent usa inferência global entre regiões. Podemos processar solicitações de entrada e resultados de saída em qualquer região comercial da AWS.

Em todos os casos, seus dados permanecem armazenados somente na região de origem da solicitação. Todos os dados transmitidos durante as operações entre regiões permanecem na rede da AWS e não atravessam a Internet pública. Criptografamos dados em trânsito entre as regiões da AWS.

Cross-Region a inferência está sempre ativada e não pode ser excluída. Para obter detalhes sobre quais regiões processam solicitações para cada recurso, consulte a seção Inferência entre regiões emthe section called “Práticas recomendadas de segurança”.

Exclusão de dados

Quando você exclui dados do AWS Security Agent:

- Os dados estão marcados para exclusão e não podem mais ser acessados por meio do serviço.
- Os dados são excluídos de todos os sistemas do AWS Security Agent em 30 dias.

Para excluir seus dados

1. No console da AWS, navegue até o AWS Security Agent.
2. Escolha os dados que você deseja excluir (análises de segurança, descobertas ou requisitos personalizados).
3. Selecione Excluir para confirmar a exclusão.

Gerenciamento de identidade e acesso para o AWS Security Agent

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. IAM os administradores controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) para usar os recursos do AWS Security Agent. IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Público

A forma como você usa AWS Identity and Access Management (IAM) difere, dependendo do trabalho que você faz no AWS Security Agent.

Usuário do serviço — Se você usa o serviço AWS Security Agent para fazer seu trabalho, seu administrador fornecerá as credenciais e as permissões de que você precisa. À medida que você usa mais recursos do AWS Security Agent para fazer seu trabalho, você pode precisar de permissões adicionais. Entender como o acesso é gerenciado pode ajudar a solicitar as permissões corretas ao administrador.

Se você não conseguir acessar um recurso no AWS Security Agent, consultethe section called “Solução de problemas de identidade e acesso ao AWS Security Agent”.

Administrador de serviços — Se você é responsável pelos recursos do AWS Security Agent em sua empresa, provavelmente tem acesso total ao AWS Security Agent. É seu trabalho determinar quais

recursos e recursos do AWS Security Agent seus usuários do serviço devem acessar. Em seguida, você deve enviar solicitações ao IAM administrador para alterar as permissões dos usuários do serviço. Revise as informações nesta página para entender os conceitos básicos do IAM. Para saber mais sobre como sua empresa pode usar o IAM AWS Security Agent, consultethe section called "Como o AWS Security Agent trabalha com IAM".

IAM administrador - Se você for IAM administrador, talvez queira saber detalhes sobre como criar políticas para gerenciar o acesso ao AWS Security Agent. Para ver exemplos de políticas baseadas em identidade do AWS Security Agent que você pode usar IAM, consulte. the section called "Exemplos de políticas baseadas em identidade do AWS Security Agent"

Autenticação com identidades

A autenticação é como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado (conectado AWS) como usuário raiz da conta da AWS Usuário do IAM, ou assumindo uma IAM função. A AWS sugere usar uma função do IAM e evitar o uso direto do usuário raiz.

Você pode entrar AWS como uma identidade federada usando credenciais fornecidas por meio de uma fonte de identidade. Centro de Identidade do AWS IAM (Centro de Identidade do IAM) usuários, a autenticação de login único da sua empresa e suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você entra como uma identidade federada, seu administrador configurou previamente a federação de identidades usando IAM funções. Ao acessar AWS usando a federação, você está assumindo indiretamente uma função.

Dependendo do tipo de usuário que você é, você pode entrar no Console de gerenciamento da AWS ou no portal de AWS acesso. Para obter mais informações sobre como fazer login AWS, consulte Como fazer login na sua conta da AWS no Guia do Sign-In usuário da AWS.

Se você acessar AWS programaticamente, AWS fornece um kit de desenvolvimento de software (SDK) e uma interface de linha de comando (CLI) para assinar criptograficamente suas solicitações usando suas credenciais. Se você não usa AWS ferramentas, você mesmo deve assinar as solicitações. Para obter mais informações sobre como usar o método recomendado para você mesmo assinar solicitações, consulte Processo de assinatura do Signature versão 4 na Referência geral da AWS.

Independentemente do método de autenticação usado, também pode ser exigido que você forneça informações adicionais de segurança. Por exemplo, AWS recomenda que você use a autenticação multifator (MFA) para aumentar a segurança da sua conta. Para saber mais, consulte Multi-factor o Guia do usuário do AWS IAM Identity Center (sucessor do AWS Single Sign-On) e o uso da autenticação multifator (MFA) na AWS no Guia do usuário do IAM.

Usuário raiz da conta da AWS

Ao criar um Conta da AWS, você começa com uma única identidade de login que tem acesso completo a todos Serviços da AWS os recursos da conta. Essa identidade é denominada usuário raiz da conta da AWS e é acessada pelo login com o endereço de e-mail e a senha que você usou para criar a conta. É altamente recomendável não usar o usuário raiz para tarefas diárias. Proteja as credenciais do usuário raiz e use-as para executar as tarefas que somente o usuário raiz possa executar. Para ver a lista completa de tarefas que exigem que você faça login como usuário raiz, consulte Tarefas que exigem credenciais de usuário raiz no Guia de referência de gerenciamento de contas.

Identidade federada

Como prática recomendada, exija que usuários humanos, incluindo usuários que precisam de acesso de administrador, usem a federação com um provedor de identidade para acessar Serviços da AWS usando credenciais temporárias.

Uma identidade federada é um usuário do seu diretório de usuários corporativo, de um provedor de identidade da web AWS Directory Service, do diretório do Identity Center ou de qualquer usuário que acesse usando credenciais fornecidas Serviços da AWS por meio de uma fonte de identidade. Quando as identidades federadas são acessadas Contas da AWS, elas assumem funções, e as funções fornecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, é recomendável usar o Centro de Identidade do AWS IAM. Você pode criar usuários e grupos no IAM Identity Center ou pode se conectar e sincronizar com um conjunto de usuários e grupos em sua própria fonte de identidade para uso em todos os seus Contas da AWS aplicativos. Para obter informações sobre o IAM Identity Center, consulte O que é o IAM Identity Center? no Guia do usuário do AWS IAM Identity Center (sucessor do AWS Single Sign-On).

Usuários do IAM e grupos

An Usuário do IAMé uma identidade dentro da sua Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, recomendamos confiar em credenciais temporárias em vez de criar Usuários do IAM credenciais de longo prazo, como senhas e chaves de acesso. No entanto, se você tiver casos de uso específicos que exijam credenciais de longo prazo Usuários do IAM, recomendamos que você alterne as chaves de acesso. Para obter mais informações, consulte Alternar as chaves de acesso regularmente para casos de uso que exijam credenciais de longo prazo no Guia do Usuário do IAM.

Um IAM grupo é uma identidade que especifica uma coleção de Usuários do IAM. Não é possível fazer login como um grupo. É possível usar grupos para especificar permissões para vários usuários de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdmins e conceder a esse grupo permissões para administrar recursos. IAM

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a uma aplicação, mas um perfil pode ser assumido por qualquer pessoa que precisar dele. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte Quando criar uma Usuário do IAM (em vez de uma função) no Guia do usuário do IAM.

IAM perfis

Uma IAM função é uma identidade dentro da sua Conta da AWS que tem permissões específicas. É semelhante a um Usuário do IAM, mas não está associado a uma pessoa específica. Você pode assumir temporariamente uma IAM função no Console de gerenciamento da AWS trocando de funções. Você pode assumir uma função chamando uma operação de AWS API AWS CLI ou usando uma URL personalizada. Para obter mais informações sobre métodos de uso de funções, consulte Como usar IAM funções no Guia do usuário do IAM.

IAM funções com credenciais temporárias são úteis nas seguintes situações:

- Acesso de usuário federado: para atribuir permissões a identidades federadas, é possível criar um perfil e definir permissões para ele. Quando uma identidade federada é autenticada, essa identidade é associada ao perfil e recebe as permissões definidas por ele. Para obter mais informações sobre perfis para federação, consulte Criar um perfil para um provedor de identidades de terceiros no Guia do usuário do IAM. Se usar o Centro de Identidade do IAM, configure um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o IAM Identity Center correlaciona o conjunto de permissões com uma função no IAM. Para obter informações sobre conjuntos de permissões, consulte Conjuntos de permissões no Guia do usuário do AWS IAM Identity Center (sucessor do AWS Single Sign-On).
- Usuário do IAM Permissões temporárias — Um Usuário do IAM pode assumir uma IAM função para assumir temporariamente diferentes permissões para uma tarefa específica.
- Cross-account acesso — Você pode usar uma IAM função para permitir que alguém (um diretor confiável) em uma conta diferente acesse recursos em sua conta. Os perfis são a principal forma de conceder acesso entre contas. No entanto, com alguns Serviços da AWS, você pode anexar uma política diretamente a um recurso (em vez de usar uma função como proxy). Para saber a

diferença entre funções e políticas baseadas em recursos para acesso entre contas, consulte Como as IAM funções diferem das políticas baseadas em recursos no Guia do usuário do IAM.

- Cross-service acesso — Alguns Serviços da AWS usam recursos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicativos Amazon EC2 ou armazene objetos em Amazon S3. Um serviço pode fazer isso usando as permissões da entidade principal de chamada, usando um perfil de serviço ou um perfil vinculado ao serviço.
  - Permissões principais — Quando você usa uma função Usuário do IAM or para realizar ações em AWS, você é considerado diretor. Permissões concedidas por políticas a uma entidade principal. Quando você usa alguns serviços, pode executar uma ação que, em seguida, acionar outra ação em outro serviço. Nesse caso, você precisa ter permissões para executar ambas as ações.
  - Função de serviço — Uma função de serviço é uma IAM função que um serviço assume para realizar ações em seu nome. Um IAM administrador pode criar, modificar e excluir uma função de serviço internamente IAM. Para obter mais informações, consulte Criar um perfil para delegar permissões a um AWS service (Serviço da AWS) no Guia do usuário do IAM.
  - Service-linked função — Uma função vinculada ao serviço é um tipo de função de serviço vinculada a uma. AWS service (Serviço da AWS) O serviço pode assumir a função de realizar uma ação em seu nome. Service-linked as funções aparecem no seu Conta da AWS e são de propriedade do serviço. Um IAM administrador pode visualizar, mas não editar, as permissões das funções vinculadas ao serviço.
- Aplicativos em execução Amazon EC2 — Você pode usar uma IAM função para gerenciar credenciais temporárias para aplicativos que estão sendo executados em uma Amazon EC2 instância e fazendo solicitações AWS CLI de AWS API. Isso é preferível a armazenar chaves de acesso na Amazon EC2 instância. Para atribuir uma AWS função a uma Amazon EC2 instância e disponibilizá-la para todos os aplicativos, você cria um perfil de instância anexado à instância. Um perfil de instância contém a função e permite que programas em execução na Amazon EC2 instância recebam credenciais temporárias. Para obter mais informações, consulte Como usar uma IAM função para conceder permissões a aplicativos executados em Amazon EC2 instâncias no Guia do usuário do IAM.

Para saber se usar IAM funções, consulte Quando criar uma IAM função (em vez de um usuário) no Guia do usuário do IAM.

Gerenciar o acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política é um objeto AWS que, quando associada a uma identidade ou recurso, define suas permissões. AWS avalia essas políticas quando um principal (usuário, usuário raiz ou sessão de função) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas é armazenada AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte Visão geral das políticas JSON no Guia do usuário do IAM.

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

Cada IAM entidade (usuário ou função) começa sem permissões. Por padrão, os usuários não podem fazer nada, nem mesmo alterar sua própria senha. Para dar permissão a um usuário para fazer algo, um administrador deve anexar uma política de permissões ao usuário. Ou o administrador pode adicionar o usuário a um grupo que tenha as permissões pretendidas. Quando um administrador concede permissões a um grupo, todos os usuários desse grupo recebem essas permissões.

IAM as políticas definem permissões para uma ação, independentemente do método usado para realizar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de função da Console de gerenciamento da AWS AWS CLI, da ou da AWS API.

Identity-based políticas

Identity-based políticas são documentos de políticas de permissões JSON que você pode anexar a uma identidade Usuário do IAM, como uma função ou grupo. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais atributos e em que condições. Para saber como criar uma política baseada em identidade, consulte Criação de IAM políticas no Guia do usuário do IAM.

Identity-based as políticas podem ser ainda categorizadas como políticas em linha ou políticas gerenciadas. As políticas em linha são anexadas diretamente a um único usuário, grupo ou perfil. As políticas gerenciadas são políticas autônomas que você pode associar a vários usuários, grupos e funções em seu Conta da AWS. As políticas AWS gerenciadas incluem políticas gerenciadas e políticas gerenciadas pelo cliente. Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte Escolher entre políticas gerenciadas e políticas em linha no Guia do Usuário do IAM.

Resource-based políticas

Resource-based políticas são documentos de política JSON que você anexa a um recurso, como um Amazon S3 bucket. Os administradores de serviços podem usar essas políticas para definir quais ações um principal específico (membro da conta, usuário ou função) pode realizar nesse recurso e sob quais condições. Resource-based as políticas são políticas em linha. Não há políticas baseadas em recursos gerenciadas.

Listas de controle de acesso (ACLs)

As listas de controle de acesso (ACLs) constituem um tipo de política que controla as entidades principais (membros, usuários ou perfis da conta) que têm permissões para acessar um recurso. As ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON. Amazon S3, AWS WAF, e Amazon VPC são exemplos de serviços que oferecem suporte a ACLs. Para saber mais sobre ACLs, consulte Visão geral da lista de controle de acesso (ACL) no Guia do desenvolvedor do Amazon Simple Storage Service.

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- Limites de permissões — Um limite de permissões é um recurso avançado no qual você define as permissões máximas que uma política baseada em identidade pode conceder a uma IAM entidade (Usuário do IAM ou função). Você pode definir um limite de permissões para uma entidade. As permissões resultantes são a interseção das políticas baseadas em identidade da entidade e seus limites de permissões. Resource-based as políticas que especificam o usuário ou a função no `Principal` campo não são limitadas pelo limite de permissões. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações sobre limites de permissões, consulte Limites de permissões para IAM entidades no Guia do usuário do IAM.
- Políticas de controle de serviço (SCPs) — SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (OU) em. AWS Organizations AWS Organizations é um serviço para agrupar e gerenciar centralmente vários Contas da AWS que sua empresa possui. Se você habilitar todos os recursos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as contas. O SCP limita as permissões para entidades em contas-membro, incluindo cada Usuário raiz da conta da AWS. Para obter mais informações sobre Organizações e SCPs, consulte Como SCPs funcionam no Guia do usuário do AWS Organizations.

- Políticas de sessão: são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para um perfil ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou da função e as políticas da sessão. As permissões também podem ser provenientes de uma política baseada em recursos. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte Políticas de sessão no Guia do usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte Lógica de avaliação de políticas no Guia do usuário do IAM.

Como o AWS Security Agent trabalha com IAM

Antes de usar IAM para gerenciar o acesso ao AWS Security Agent, saiba quais IAM recursos estão disponíveis para uso com o AWS Security Agent.

Recurso do IAM	Suporte ao AWS Security Agent
the section called “Identity-based políticas para o AWS Security Agent”	Sim
the section called “Resource-based políticas dentro do AWS Security Agent”	Não
the section called “Ações políticas para o AWS Security Agent”	Sim
the section called “Recursos de política para o AWS Security Agent”	Parcial
the section called “Chaves de condição de política para o AWS Security Agent”	Sim
the section called “Listas de controle de acesso (ACLs) no AWS Security Agent”	Não

Recurso do IAM	Suporte ao AWS Security Agent
the section called “Attribute-based controle de acesso (ABAC) com o AWS Security Agent”	Não
the section called “Usando credenciais temporárias com o AWS Security Agent”	Sim
the section called “Encaminhe sessões de acesso para o AWS Security Agent”	Sim
the section called “Funções de serviço do AWS Security Agent”	Não
the section called “Service-linked funções do AWS Security Agent”	Sim

Para obter uma visão de alto nível de como o AWS Security Agent e outros Serviços da AWS trabalham com IAM, consulte Serviços da AWS esse trabalho IAM no Guia do usuário do IAM.

Identity-based políticas para o AWS Security Agent

Compatível com políticas baseadas em identidade: sim

Identity-based políticas são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como um usuário do IAM, um grupo de usuários ou uma função. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais atributos e em que condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

Com políticas IAM baseadas em identidade, você pode especificar ações e recursos permitidos ou negados, bem como as condições sob as quais as ações são permitidas ou negadas. Você não pode especificar a entidade principal em uma política baseada em identidade porque ela se aplica ao usuário ou perfil ao qual ela está anexada. Para saber mais sobre todos os elementos que você usa em uma política JSON, consulte a referência aos elementos da política IAM JSON no Guia do usuário do IAM.

Identity-based exemplos de políticas para o AWS Security Agent

Para ver exemplos de políticas baseadas em identidade do AWS Security Agent, consulte. the section called “Exemplos de políticas baseadas em identidade do AWS Security Agent”

Resource-based políticas dentro do AWS Security Agent

Compatibilidade com políticas baseadas em recursos: não

Resource-based políticas são documentos de política JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o atributo ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse atributo e em que condições. É necessário especificar uma entidade principal em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou serviços da AWS.

Para permitir o acesso entre contas, você pode especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em recursos. Adicionar uma entidade principal entre contas à política baseada em recurso é apenas metade da tarefa de estabelecimento da relação de confiança. Quando o principal e o recurso estão em contas da AWS diferentes, um administrador do IAM na conta confiável também deve conceder permissão à entidade principal (usuário ou função) para acessar o recurso. Eles concedem permissão ao anexar uma política baseada em identidade para a entidade. No entanto, se uma política baseada em recurso conceder acesso a uma entidade principal na mesma conta, nenhuma política baseada em identidade adicional será necessária. Para obter mais informações, consulte Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Ações políticas para o AWS Security Agent

Suporta ações Sim

Os administradores podem usar as políticas JSON da AWS para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O `Action` elemento de uma política IAM baseada em identidade descreve a ação ou ações específicas que serão permitidas ou negadas pela política. As ações de política geralmente têm o mesmo nome da operação de AWS API associada. A ação é usada em uma política para conceder permissões para executar a operação associada.

As ações de política no AWS Security Agent usam o seguinte prefixo antes da ação:`securityagent:`. Por exemplo, para conceder permissão a alguém para criar um ambiente com a operação da `CreateEnvironment` API do AWS Security Agent, você inclui a `securityagent:CreateEnvironment` ação na política dessa pessoa. As instruções de política devem incluir um elemento `Action` ou `NotAction`. O AWS Security Agent define seu próprio conjunto de ações que descrevem tarefas que você pode realizar com esse serviço.

Para especificar várias ações em uma única instrução, separe-as com vírgulas, como segue:

```
"Action": [
      "securityagent:action1",
      "securityagent:action2"
```

Você também pode especificar várias ações usando caracteres curinga (*). Por exemplo, para especificar todas as ações que começam com a palavra `List`, inclua a seguinte ação:

```
"Action": "securityagent:List*"
```

Recursos de política para o AWS Security Agent

Suporte a recursos de políticas: parcial

Os administradores podem usar as políticas JSON da AWS para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou `NotResource`. Como prática recomendada, especifique um recurso usando seu nome do recurso da Amazon (ARN). Isso pode ser feito para ações que oferecem compatibilidade com um tipo de recurso específico, conhecido como permissões em nível de recurso.

Para ações que não oferecem suporte a permissões em nível de recurso, como operações de listagem, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

```
"Resource": "*"
```

Algumas ações da API do AWS Security Agent oferecem suporte a vários recursos. Por exemplo, vários ambientes podem ser referenciados ao chamar a ação da `ListEnvironments` API. Para especificar vários recursos em uma única declaração, separe os ARNs com vírgulas.

```
"Resource": [
      "EXAMPLE-RESOURCE-1",
      "EXAMPLE-RESOURCE-2"
```

Por exemplo, o recurso de ambiente do AWS Security Agent tem o seguinte ARN:

```
arn:${Partition}:securityagent:${Region}:${Account}:environment/${EnvironmentId}
```

Para especificar os ambientes `my-environment-1` e `my-environment-2` em sua declaração, use os seguintes exemplos de ARNs:

```
"Resource": [
        "arn:aws:securityagent:us-east-1:123456789012:environment/my-environment-1",
        "arn:aws:securityagent:us-east-1:123456789012:environment/my-environment-2"
```

Para especificar todos os ambientes que pertencem a uma conta específica, use o caractere curinga (*):

```
"Resource": "arn:aws:securityagent:us-east-1:123456789012:environment/*"
```

Chaves de condição de política para o AWS Security Agent

Compatível com chaves de condição de política específicas de serviço: sim

Os administradores podem usar as políticas JSON da AWS para especificar quem tem acesso a quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O `Condition` elemento (ou `Condition` bloco) permite especificar as condições nas quais uma declaração está em vigor. O elemento `Condition` é opcional. É possível criar expressões condicionais que usem agentes de condição, como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação.

Se você especificar vários elementos de `Condition` em uma declaração ou várias chaves em um único elemento de `Condition`, a AWS os avaliará usando uma operação lógica `AND`. Se você especificar vários valores para uma única chave de condição, AWS avalia a condição usando uma `OR` operação lógica. Todas as condições devem ser atendidas para que as permissões da instrução sejam concedidas.

Você também pode usar variáveis de espaço reservado ao especificar condições. Por exemplo, você pode conceder uma Usuário do IAM permissão para acessar um recurso somente se ele estiver

marcado com o Usuário do IAM nome dele. Para obter mais informações, consulte elementos IAM de política: variáveis e tags no Guia do usuário do IAM.

O AWS Security Agent define seu próprio conjunto de chaves de condição e também oferece suporte ao uso de algumas chaves de condição globais. Para ver todas as chaves de condição AWS globais, consulte as chaves de contexto de condição AWS global no Guia do usuário do IAM.

Listas de controle de acesso (ACLs) no AWS Security Agent

Compatível com ACLs: não

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

Attribute-based controle de acesso (ABAC) com o AWS Security Agent

Oferece compatibilidade com ABAC (tags em políticas): não

Usando credenciais temporárias com o AWS Security Agent

Compatível com credenciais temporárias: sim

Alguns serviços da AWS não funcionam quando você faz login usando credenciais temporárias. Para obter informações adicionais, incluindo quais serviços da AWS funcionam com credenciais temporárias, consulte Serviços da AWS que funcionam com o IAM no Guia do usuário do IAM.

Você está usando credenciais temporárias se fizer login no AWS Management Console usando qualquer método, exceto um nome de usuário e senha. Por exemplo, quando você acessa a AWS usando o link de login único (SSO) da sua empresa, esse processo cria automaticamente credenciais temporárias. Você também cria automaticamente credenciais temporárias quando faz login no console como usuário e, em seguida, alterna perfis. Para obter mais informações sobre como alternar funções, consulte Alternar para um perfil do IAM (console) no Guia do usuário do IAM.

Você pode criar manualmente credenciais temporárias usando a AWS CLI ou a API da AWS. Em seguida, você pode usar essas credenciais temporárias para acessar a AWS. A AWS recomenda que você gere dinamicamente credenciais temporárias em vez de usar chaves de acesso de longo prazo. Para obter mais informações, consulte Credenciais de segurança temporárias no IAM.

Encaminhe sessões de acesso para o AWS Security Agent

Compatibilidade com o recurso de encaminhamento de sessões de acesso (FAS): sim

Quando você usa um usuário ou uma função do IAM para realizar ações na AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O FAS usa as permissões do diretor que chama um serviço da AWS, combinadas com o serviço da AWS solicitante para fazer solicitações aos serviços downstream. As solicitações do FAS só são feitas quando um serviço recebe uma solicitação que exige interações com outros serviços ou recursos da AWS para ser concluída. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte Sessões de acesso direto.

Funções de serviço do AWS Security Agent

Compatível com perfis de serviço: não

O perfil de serviço é um perfil do IAM que um serviço assume para executar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte Criar uma função para delegar permissões a um serviço da AWS no Guia do usuário do IAM.

Service-linked funções do AWS Security Agent

Compatibilidade com perfis vinculados a serviços: sim

Uma função vinculada a serviços é um tipo de função de serviço vinculada a um serviço da AWS. O serviço pode assumir a função de realizar uma ação em seu nome. Service-linked as funções aparecem na sua conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para perfis vinculados ao serviço.

Exemplos de políticas baseadas em identidade do AWS Security Agent

Por padrão, Usuários do IAM as funções não têm permissão para criar ou modificar recursos do AWS Security Agent. Eles também não podem realizar tarefas usando a AWS API Console de gerenciamento da AWS AWS CLI, ou. Um IAM administrador deve criar IAM políticas que concedam aos usuários e funções permissão para realizar operações de API específicas nos recursos especificados de que precisam. O administrador deve então anexar essas políticas aos grupos Usuários do IAM ou grupos que exigem essas permissões.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documentos de política JSON, consulte Como criar políticas usando o editor JSON no Guia do usuário do IAM.

Práticas recomendadas de política

Identity-based as políticas determinam se alguém pode criar, acessar ou excluir recursos do AWS Security Agent em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas AWS gerenciadas e avance para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para obter mais informações, consulte políticas AWS gerenciadas ou políticas gerenciadas pela AWS para funções de trabalho no Guia do usuário do IAM.
- Aplique permissões com privilégios mínimos — Ao definir permissões com IAM políticas, conceda somente as permissões necessárias para realizar uma tarefa. Você faz isso definindo as ações que podem ser executadas em recursos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para obter mais informações sobre como usar IAM para aplicar permissões, consulte Políticas e permissões IAM no Guia do usuário do IAM.
- Use condições nas IAM políticas para restringir ainda mais o acesso — Você pode adicionar uma condição às suas políticas para limitar o acesso a ações e recursos. Por exemplo, é possível escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como CloudFormation. Para obter mais informações, consulte Elementos da política IAM JSON: condição no Guia do usuário do IAM.
- Use IAM Access Analyzer para validar suas IAM políticas para garantir permissões seguras e funcionais — IAM Access Analyzer valida políticas novas e existentes para que as políticas sigam a linguagem de políticas (JSON) e IAM as melhores práticas. IAM IAM Access Analyzer fornece mais de 100 verificações de políticas e recomendações práticas para ajudá-lo a criar políticas seguras e funcionais. Para obter mais informações, consulte a validação de IAM Access Analyzer políticas no Guia do usuário do IAM.
- Exigir autenticação multifator (MFA) — Se você tiver um cenário que Usuários do IAM exija ou faça root de usuários em sua conta, ative a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para obter mais informações, consulte Como configurar o acesso à MFA-protected API no Guia do usuário do IAM.

Usando o console do AWS Security Agent

Para acessar o console do AWS Security Agent, um diretor do IAM deve ter um conjunto mínimo de permissões. Essas permissões devem permitir que o diretor liste e visualize detalhes sobre os recursos do AWS Security Agent em seu Conta da AWS. Se você criar uma política baseada em identidade que seja mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para as entidades principais com essa política anexada.

Para garantir que seus diretores do IAM ainda possam usar o console do AWS Security Agent, crie uma política com seu próprio nome exclusivo. Anexe a política às entidades principais. Para obter mais informações, consulte Adição de permissões a um usuário no Guia do usuário do IAM.

Para obter detalhes da política, consultethe section called “Política gerenciada pela AWS: SecurityAgentWebAppAPIPolicy”.

Você não precisa permitir permissões mínimas do console para usuários que estão fazendo chamadas somente para a API AWS CLI ou para a AWS API. Em vez disso, permita o acesso somente às ações que correspondem à operação da API que você está tentando executar.

Solução de problemas de identidade e acesso ao AWS Security Agent

Use as informações a seguir para ajudá-lo a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com o AWS Security Agent e. IAM

AccessDeniedException

Se você receber um `AccessDeniedException` ao chamar uma operação de API da AWS, as credenciais principais do IAM que você está usando não têm as permissões necessárias para fazer essa chamada.

```
An error occurred (AccessDeniedException) when calling the CreateEnvironment operation:
User: arn:aws:iam::111122223333:user/user_name is not authorized to perform:
securityagent:CreateEnvironment on resource:
 arn:aws:securityagent:region:111122223333:environment/my-env
```

Na mensagem de exemplo anterior, o usuário não tem permissão para chamar a operação de `CreateEnvironment` API do AWS Security Agent. Para fornecer permissões de administrador do AWS Security Agent a um diretor do IAM, consultethe section called “Exemplos de políticas baseadas em identidade do AWS Security Agent”.

Para obter mais informações gerais sobre o IAM, consulte Controle o acesso aos recursos da AWS usando políticas no Guia do usuário do IAM.

Quero permitir que pessoas fora da minha Conta da AWS para acessar meus recursos do AWS Security Agent

É possível criar um perfil que os usuários de outras contas ou pessoas fora da organização podem usar para acessar seus recursos. É possível especificar quem é confiável para assumir o perfil. Para serviços que oferecem compatibilidade com políticas baseadas em recursos ou listas de controle de acesso (ACLs), é possível usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o AWS Security Agent oferece suporte a esses recursos, consultethe section called "Como o AWS Security Agent trabalha com IAM".
- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte Como fornecer acesso a um Usuário do IAM em outro Conta da AWS de sua propriedade no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Como fornecer acesso Contas da AWS a terceiros no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte Conceder acesso a usuários autenticados externamente (federação de identidades) no Guia do usuário do IAM.
- Para saber a diferença entre usar funções e políticas baseadas em recursos para acesso entre contas, consulte Como as IAM funções diferem das políticas baseadas em recursos no Guia do usuário do IAM.

Resposta a incidentes

O AWS Security Agent é um serviço proativo de testes de segurança projetado para identificar e evitar vulnerabilidades antes que elas possam ser exploradas. O serviço se concentra na validação preventiva da segurança por meio de análises de design, análise de código e testes de penetração, em vez de detecção ou resposta reativa a incidentes.

Detecção de incidente

O AWS Security Agent não fornece recursos de detecção de incidentes. O serviço opera como uma ferramenta proativa de teste de segurança que valida os aplicativos em busca de vulnerabilidades

durante o desenvolvimento e antes da implantação. Para monitoramento de segurança em tempo de execução e detecção de incidentes, use serviços como Amazon GuardDuty, AWS Security Hub ou Amazon CloudWatch.

Alerta de incidentes

O AWS Security Agent não gera alertas em tempo real para incidentes de segurança. O serviço fornece descobertas de segurança por meio do AWS Console após concluir análises de design, análises de código ou testes de penetração. Essas descobertas representam possíveis vulnerabilidades descobertas durante os testes, em vez de incidentes de segurança ativos.

Remediação de incidentes

O AWS Security Agent não fornece remediação automática de incidentes. O serviço identifica vulnerabilidades de segurança e fornece diretrizes de remediação, incluindo:

- Descrições detalhadas das vulnerabilidades identificadas
- Caminhos de exploração reproduzíveis para descobertas validadas
- Correções de código específicas e diretrizes de implementação
- Análise de impacto para problemas descobertos

As equipes de desenvolvimento e segurança usam essa orientação para tratar manualmente as vulnerabilidades antes que elas cheguem aos ambientes de produção.

Apoiando atividades de resposta a incidentes

Embora o AWS Security Agent não tenha sido projetado para responder a incidentes, as equipes de segurança podem usar o serviço para apoiar atividades pós-incidentes:

validação de vulnerabilidade

: Depois de um incidente de segurança, use o AWS Security Agent para testar se existem vulnerabilidades semelhantes em outros aplicativos ou ambientes.

Avaliação da postura de segurança

: Realize testes de penetração para validar as melhorias de segurança implementadas como parte da remediação de incidentes.

Análise da causa raiz

Use os recursos de análise de segurança de código para identificar como vulnerabilidades semelhantes podem existir em outras bases de código.

Validação de conformidade para o AWS Security Agent

Para saber se um serviço da AWS está dentro do escopo de programas de conformidade específicos, consulte Serviços da AWS em Escopo por Programa de Conformidade e escolha o programa de conformidade no qual você está interessado. Para obter informações gerais, consulte Programas de conformidade da AWS.

Você pode fazer download de relatórios de auditoria de terceiros usando o Artefato da AWS. Para obter mais informações, consulte Download de relatórios no AWS Artifact.

Sua responsabilidade de conformidade ao usar os serviços da AWS é determinada pela confidencialidade dos seus dados, pelos objetivos de conformidade da sua empresa e pelas leis e regulamentações aplicáveis. A fornece os seguintes recursos para ajudar com a conformidade:

- Governança e conformidade de segurança: esses guias de implementação de solução abordam considerações sobre a arquitetura e fornecem etapas para implantar recursos de segurança e conformidade.
- Referência de serviços qualificados para HIPAA: lista os serviços qualificados para HIPAA. Nem todos os serviços da AWS estão qualificados para a HIPAA.
- Recursos de conformidade da AWS: esta coleção de guias e pastas de trabalho pode ser aplicada ao seu setor e local.
- Guias de conformidade do cliente da AWS — Entenda o modelo de responsabilidade compartilhada sob o prisma da conformidade. Os guias resumem as melhores práticas para proteger os serviços da AWS e mapeiam a orientação dos controles de segurança em várias estruturas (incluindo o Instituto Nacional de Padrões e Tecnologia (NIST), o Conselho de Padrões de Segurança do Setor de Cartões de Pagamento (PCI) e a Organização Internacional de Padronização (ISO)).
- Avaliar recursos com regras no Guia do desenvolvedor do AWS Config: o serviço AWS Config avalia como as configurações de recursos estão em conformidade com práticas internas, diretrizes do setor e regulamentos.
- AWS Security Hub — Esse serviço da AWS fornece uma visão abrangente do seu estado de segurança na AWS. O Security Hub usa controles de segurança para avaliar seus recursos da

AWS e verificar sua conformidade com os padrões e melhores práticas do setor de segurança. Para obter uma lista dos serviços e controles aceitos, consulte a Referência de controles do Security Hub.

- Amazon GuardDuty — Esse serviço da AWS detecta possíveis ameaças às suas contas, cargas de trabalho, contêineres e dados da AWS monitorando seu ambiente em busca de atividades suspeitas e maliciosas. GuardDuty pode ajudá-lo a atender a vários requisitos de conformidade, como o PCI DSS, atendendo aos requisitos de detecção de intrusões exigidos por determinadas estruturas de conformidade.
- AWS Audit Manager: esse serviço da AWS ajuda a auditar continuamente o uso da AWS para simplificar a forma como você gerencia os riscos e a conformidade com regulamentos e padrões do setor.

Resiliência no AWS Security Agent

> **Note**
>
> O AWS Security Agent está disponível nas seguintes regiões: Leste dos EUA (Norte da Virgínia) —`us-east-1`, Oeste dos EUA (Oregon) —`us-west-2`, Ásia-Pacífico (Mumbai) —`ap-south-1`, Ásia-Pacífico (Cingapura) —`ap-southeast-1`, Ásia-Pacífico (Sydney) —`ap-southeast-2`, Ásia-Pacífico (Tóquio) —`ap-northeast-1`, Europa (Frankfurt) —`eu-central-1`, Europa (Irlanda) — `eu-west-1` e América do Sul (São Paulo) —. `sa-east-1` Ele usa inferência entre regiões. Na Ásia-Pacífico (Mumbai), Ásia-Pacífico (Cingapura) e América do Sul (São Paulo), o AWS Security Agent usa inferência global entre regiões, que encaminha solicitações de inferência para qualquer região comercial da AWS. Em todas as outras regiões, ele usa inferência geográfica entre regiões, o que mantém o processamento de dados dentro de limites geográficos específicos. O AWS Security Agent é uma ferramenta usada durante o desenvolvimento do seu aplicativo e não deve ser implantado como uma infraestrutura crítica ou voltada para o cliente.

A infraestrutura global da AWS é criada com base em regiões e zonas de disponibilidade da AWS. As regiões da AWS fornecem várias zonas de disponibilidade separadas e fisicamente isoladas, que são conectadas com redes de baixa latência, throughput alto e altamente redundantes. Com as zonas de disponibilidade, é possível projetar e operar aplicações e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade

são altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre as regiões e as zonas de disponibilidade da AWS, consulte Infraestrutura global da AWS.

Segurança da infraestrutura no AWS Security Agent

Como um serviço gerenciado, o AWS Security Agent é protegido pela segurança de rede global da AWS. Para obter informações sobre os serviços de segurança da AWS e como a AWS protege a infraestrutura, consulte Segurança na nuvem da AWS. Para projetar seu ambiente da AWS usando as melhores práticas de segurança da infraestrutura, consulte Segurança na Well-Architected estrutura da AWS.

Isolamento de rede

O AWS Security Agent é um serviço totalmente gerenciado acessado por meio do console da AWS e do aplicativo web do AWS Security Agent. O acesso ao serviço é controlado por meio do AWS Identity and Access Management (IAM) ou do AWS IAM Identity Center, que pode ser integrado ao seu provedor de identidade.

O AWS Security Agent oferece suporte a endpoints de VPC de interface desenvolvidos pela AWS PrivateLink, permitindo que você acesse de forma privada as APIs de serviço da sua VPC sem atravessar a Internet pública. Para obter mais informações, consulte the section called “Endpoints da VPC de interface”.

O AWS Security Agent exige acesso à Internet para realizar testes de penetração em aplicativos de destino e para operações do plano de controle. O serviço usa a AWS-managed infraestrutura para testes e não provisiona instâncias do EC2 ou outros recursos computacionais em sua conta. Se você configurar o acesso à VPC para testes de penetração, o Security Agent cria uma ENI na sua sub-rede, mas essa ENI não tem um endereço IP público. Para obter mais informações, consulte the section called “Conecte o agente aos recursos privados da VPC”.

Multi-tenancy e isolamento de recursos

O AWS Security Agent é um serviço multilocatário. Análises de segurança, descobertas e dados de clientes são isolados em contas individuais da AWS e criptografados em repouso. A AWS aplica controles padrão de isolamento de infraestrutura para garantir que as atividades de teste de segurança de um cliente não afetem o desempenho ou a confidencialidade de outro cliente.

AWS Agente de segurança e endpoints VPC de interface (AWS PrivateLink)

Você pode usar AWS PrivateLink para criar uma conexão privada entre sua VPC e o AWS Security Agent. Você pode acessar o Security Agent como se estivesse em sua VPC, sem o uso de um gateway de internet, dispositivo NAT, conexão VPN ou conexão Direct Connect. As instâncias em sua VPC não precisam de endereços IP públicos para acessar o Security Agent.

Estabeleça essa conectividade privada criando um endpoint de interface, habilitado pelo AWS PrivateLink. Criaremos um endpoint de interface de rede em cada sub-rede que você habilitar para o endpoint de interface. Essas são interfaces de rede gerenciadas pelo solicitante que servem como ponto de entrada para o tráfego destinado ao Security Agent.

Para obter mais informações, consulte Acesse os AWS serviços AWS PrivateLink no AWS PrivateLink Guia.

Considerações sobre o Security Agent

Antes de configurar um endpoint de interface para o Security Agent, revise as considerações no AWS PrivateLink Guia.

O Security Agent oferece suporte para fazer chamadas para todas as suas ações de API a partir de sua VPC, incluindo operações para gerenciar espaços de agentes, testes de penetração e descobertas de segurança.

Você pode selecionar IPv4, IPv6 ou dualstack ao criar um endpoint.

As políticas de VPC endpoint são compatíveis com o Security Agent. Por padrão, o acesso total ao Security Agent é permitido por meio do endpoint da interface. Você pode controlar o acesso anexando uma política de endpoint ao endpoint da interface ou associando um grupo de segurança às interfaces de rede do endpoint.

Todas as chamadas de API para o Security Agent são criptografadas usando TLS 1.2 ou superior, incluindo chamadas feitas por meio de VPC endpoints. Para mais informações sobre a proteção de dados, consulte the section called “Proteção de dados”. As chamadas da API do Security Agent são registradas AWS CloudTrail. Para obter mais informações, consulte the section called “Exemplo: entradas do arquivo de log do AWS Security Agent”.

Crie um endpoint de interface para o Security Agent

Você pode criar um endpoint de interface para o Security Agent usando o console Amazon VPC ou AWS a Interface de Linha de Comando (AWS CLI). Para obter mais informações, consulte Criar um endpoint de interface no AWS PrivateLink Guia.

Crie um endpoint de interface para o Security Agent usando o seguinte nome de serviço:

```
com.amazonaws.<region>.securityagent
```

Para criar o endpoint da interface usando a AWS CLI, execute o comando a seguir. Substitua a região, o ID da VPC, os IDs de sub-rede e o ID do grupo de segurança pelos seus próprios valores.

```
aws ec2 create-vpc-endpoint \
    --vpc-id vpc-1a2b3c4d \
    --vpc-endpoint-type Interface \
    --service-name com.amazonaws.<region>.securityagent \
    --subnet-ids subnet-1a2b3c4d subnet-5e6f7a8b \
    --security-group-ids sg-1a2b3c4d \
    --private-dns-enabled
```

Important

O DNS privado deve estar habilitado para o endpoint da interface. O Security Agent exige que você use o nome DNS regional padrão (por exemplo,`securityagent.us-east-1.api.aws`) para fazer solicitações de API por meio do endpoint. Não há suporte para chamar diretamente o URL do VPCE específico do endpoint.

Para usar o DNS privado, você deve definir os `enableDnsHostnames` atributos `enableDnsSupport` e `true` para sua VPC. Para acessar mais informações, consulte Atributos de DNS para sua VPC no Guia do usuário da Amazon VPC.

Configurar grupos de segurança para o endpoint da interface

Ao criar um endpoint de interface, você pode associar grupos de segurança às interfaces de rede do endpoint para controlar o tráfego para o Security Agent. Crie um grupo de segurança que permita tráfego HTTPS de entrada (porta 443) dos recursos em sua VPC que precisam se comunicar com o Security Agent.

O grupo de segurança deve incluir a seguinte regra de entrada:

- Tipo: HTTPS
- Protocolo: TCP
- Alcance de portas: 443
- Fonte: Seu bloco CIDR da VPC (por exemplo,`10.0.0.0/16`) ou os IDs do grupo de segurança dos recursos que precisam de acesso ao Security Agent

Para obter mais informações, consulte Grupos de segurança no AWS PrivateLink Guia.

Criar uma política de endpoint para o endpoint de interface

Uma política de endpoint é um recurso do IAM que pode ser anexado ao endpoint de interface. A política de endpoint padrão permite acesso total ao Security Agent por meio do endpoint da interface. Para controlar o acesso permitido ao Security Agent a partir de sua VPC, anexe uma política de endpoint personalizada ao endpoint da interface.

Uma política de endpoint especifica as seguintes informações:

- Os diretores que podem realizar ações (AWS contas, usuários do IAM e funções do IAM).
- As ações que podem ser realizadas.
- Os recursos nos quais as ações podem ser executadas.

Para obter mais informações, consulte Controlar o acesso aos serviços usando políticas de endpoint no AWS PrivateLink Guia.

Exemplo: política de VPC endpoint para ações do Security Agent AWS

Veja a seguir um exemplo de uma política de endpoint para o AWS Security Agent. Quando anexada a um endpoint, essa política concede acesso às ações listadas do AWS Security Agent para todos os diretores em todos os recursos.

```
{
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": "*",
         "Action": [
```

```
            "securityagent:CreatePentest",
            "securityagent:ListPentests",
            "securityagent:BatchGetPentests",
            "securityagent:StartPentestExecution",
            "securityagent:StopPentestExecution",
            "securityagent:ListFindings",
            "securityagent:BatchGetFindings"
         ],
         "Resource": "*"
      }
   ]
}
```

Exemplo: política de VPC endpoint que nega todo o acesso de uma conta especificada AWS

A política de VPC endpoint a seguir nega à AWS conta `123456789012` todo o acesso aos recursos que usam o endpoint. A política permite todas as ações de outras contas.

```
{
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": "*",
         "Action": "securityagent:*",
         "Resource": "*"
      },
      {
         "Effect": "Deny",
         "Principal": {
            "AWS": "123456789012"
         },
         "Action": "securityagent:*",
         "Resource": "*"
      }
   ]
}
```

Solução de problemas

A conexão expira ao chamar a API do Security Agent

- Verifique se o status do VPC endpoint é: `available`

```
aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-1a2b3c4d \
    --query "VpcEndpoints[].State"
```

- Confirme se o grupo de segurança associado ao endpoint permite tráfego TCP de entrada na porta 443 do seu CIDR de VPC ou do grupo de segurança de origem.
- Verifique se as tabelas de rotas da VPC não roteiam o tráfego do Security Agent para um gateway da Internet ou gateway NAT em vez do endpoint.

A resolução de DNS falha para **`securityagent.<region>.api.aws`**

- Verifique se o DNS privado está habilitado no endpoint:

```
aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-1a2b3c4d \
    --query "VpcEndpoints[].PrivateDnsEnabled"
```

- Confirme se ambos `enableDnsSupport` `enableDnsHostnames` estão configurados `true` em sua VPC:

```
aws ec2 describe-vpc-attribute --vpc-id vpc-1a2b3c4d --attribute enableDnsSupport
aws ec2 describe-vpc-attribute --vpc-id vpc-1a2b3c4d --attribute enableDnsHostnames
```

`Access denied`erros ao chamar as APIs do Security Agent

- Verifique a política de VPC endpoint para garantir que ela permita as ações que você está tentando realizar.
- Verifique se sua política de identidade do IAM concede `securityagent:` as permissões necessárias.
- Se estiver usando uma política de endpoint personalizada, confirme se tanto a política de endpoint quanto a política do IAM permitem a solicitação.

Análise de configuração e vulnerabilidade no AWS Security Agent

A configuração e os controles de TI são uma responsabilidade compartilhada entre a AWS e você, nosso cliente. Para obter mais informações, consulte o modelo de responsabilidade compartilhada da AWS.

Cross-service prevenção delegada confusa

O problema de confused deputy é uma questão de segurança em que uma entidade que não tem permissão para executar uma ação pode coagir uma entidade mais privilegiada a executar a ação. Na AWS, a falsificação de identidade entre serviços pode resultar em um problema confuso de delegado. Cross-service a representação pode ocorrer quando um serviço (o serviço de chamada) chama outro serviço (o serviço chamado). O serviço de chamada pode ser manipulado a usar suas permissões para atuar nos recursos de outro cliente de modo a não ter permissão de acesso. Para evitar isso, a AWS fornece ferramentas que ajudam você a proteger seus dados para todos os serviços com diretores de serviços que receberam acesso aos recursos em sua conta. Para obter mais informações, consulte Cross-service Confused Deputy Prevention no Guia do usuário do AWS Identity and Access Management.

Melhores práticas de segurança para o AWS Security Agent

O AWS Security Agent fornece vários recursos de segurança a serem considerados ao desenvolver e implementar suas próprias políticas de segurança. As práticas recomendadas a seguir são diretrizes gerais e não representam uma solução completa de segurança. Como essas práticas recomendadas podem não ser adequadas ou suficientes para o seu ambiente, trate-as como considerações úteis em vez de prescrições.

Use ambientes que não sejam de produção para testes de penetração

O AWS Security Agent usa um conjunto abrangente de ferramentas de teste de penetração da distribuição Kali Linux. Essas ferramentas são projetadas para identificar vulnerabilidades de segurança e podem realizar ações que modificam o estado do aplicativo, os dados ou as configurações do sistema.

Prática recomendada: realize testes de penetração em ambientes que não sejam de produção que reflitam sua configuração de produção. Esses ambientes de teste devem:

- Não contém dados ativos do cliente ou informações confidenciais de produção
- Seja isolado dos sistemas de produção
- Tenha configurações e controles de segurança semelhantes aos da produção
- Não usar credenciais com acesso aos sistemas de produção

Testes em ambientes de produção podem resultar em:

- Modificação ou exclusão de dados
- Interrupções no serviço ou degradação do desempenho
- Mudanças de estado não intencionais
- Acionamento de alertas de segurança ou procedimentos de resposta a incidentes

Valide as descobertas AI-generated de segurança

O AWS Security Agent realiza análises de segurança usando agentes de IA. Devido à natureza não determinística dos sistemas de IA, os testes de penetração podem produzir resultados variados em diferentes execuções.

Prática recomendada: valide as descobertas de segurança antes de tomar medidas corretivas:

- Analise os scripts de verificação gerados pelo AWS Security Agent para cada descoberta
- Execute scripts de verificação em seu ambiente de teste para confirmar a vulnerabilidade
- Considere a realização de vários testes de penetração para garantir uma cobertura abrangente
- Aplique um julgamento profissional de segurança para avaliar a gravidade da descoberta e a capacidade de exploração

Nem todos os problemas identificados podem representar vulnerabilidades exploráveis em seu contexto específico de implantação.

Revise e teste o código de remediação gerado

O AWS Security Agent pode gerar correções de código e melhorias de segurança para vulnerabilidades identificadas. Essas AI-generated correções exigem verificação antes da implantação.

Prática recomendada: analise todas as alterações de código geradas:

- Examine as correções propostas para verificar se estão completas e corretas
- Teste minuciosamente as correções em ambientes que não sejam de produção
- Verifique se as correções não introduzem novas vulnerabilidades nem interrompem a funcionalidade
- Use o AWS Security Agent para testar novamente após aplicar as correções ou executar os scripts de verificação fornecidos

- Siga os processos de revisão e aprovação de código da sua organização

Acesso ao repositório de código

O AWS Security Agent pode fornecer orientação de segurança sobre alterações de código por meio de comentários de pull request e integração de revisão de código. Para proteger informações confidenciais de segurança, essa funcionalidade opera sob restrições específicas.

Limitação: as diretrizes de segurança de código são restritas somente a repositórios privados. Isso garante que:

- As descobertas de segurança permanecem confidenciais para sua organização
- As possíveis vulnerabilidades não são divulgadas publicamente antes da correção
- As recomendações de correção geradas não expõem detalhes de exploração

O AWS Security Agent não fornece diretrizes de segurança de código para repositórios públicos. Não comentará sobre repositórios públicos ou projetos de código aberto nos quais as descobertas de segurança seriam visíveis publicamente.

Os testes de penetração do AWS Security Agent podem analisar e corrigir repositórios públicos e privados que você configurou para o pentest. Se o repositório for público, o código de correção será fornecido como um arquivo diff para download em vez de uma pull request.

URLs acessíveis

Os URLs acessíveis especificam endpoints adicionais que o ambiente de teste de penetração pode acessar durante o teste. Eles são necessários quando seu aplicativo depende de serviços externos, como provedores de autenticação terceirizados ou CDNs. Todas as dependências de rede necessárias para o teste devem ser especificadas como URLs de destino ou URLs acessíveis. A rede bloqueia o acesso a qualquer endpoint não especificado.

Implicações de segurança: o AWS Security Agent não está instruído a realizar testes de segurança em URLs acessíveis. Ao especificar URLs acessíveis, você indica confiança nessas dependências. Os dados do teste de penetração, incluindo credenciais, podem ser transmitidos para esses endpoints de URL acessíveis durante o teste.

Inferência entre regiões

O AWS Security Agent seleciona automaticamente a região ideal para processar suas solicitações de inferência. Isso maximiza os recursos computacionais disponíveis, a disponibilidade do modelo e oferece a melhor experiência ao cliente. Seus dados permanecem armazenados somente na região de origem da solicitação; no entanto, as solicitações de entrada e os resultados de saída podem ser processados fora dessa região. Transmitimos todos os dados criptografados pela rede AWS.

O AWS Security Agent usa dois tipos de inferência entre regiões, dependendo da região:

- Inferência geográfica entre regiões — mantém o processamento de dados dentro de limites geográficos específicos (como EUA, UE, Austrália ou Japão) para a maioria dos recursos. Para remediação de código, as solicitações da Austrália e do Japão são processadas na União Europeia. Usado no Leste dos EUA (Norte da Virgínia) —`us-east-1`, Oeste dos EUA (Oregon) —`us-west-2`, Ásia-Pacífico (Sydney) —`ap-southeast-2`, Ásia-Pacífico (Tóquio) —`ap-northeast-1`, Europa (Frankfurt) — `eu-central-1` e Europa (Irlanda) —. `eu-west-1`
- Inferência global entre regiões — encaminha as solicitações de inferência para qualquer região comercial da AWS, otimizando os recursos disponíveis e permitindo uma maior taxa de transferência do modelo. Usado na Ásia-Pacífico (Mumbai) —`ap-south-1`, Ásia-Pacífico (Cingapura) — `ap-southeast-1` e América do Sul (São Paulo) —. `sa-east-1`

Para regiões que usam inferência global entre regiões, as solicitações de entrada e os resultados de saída podem ser processados em qualquer região comercial da AWS. Todos os dados transmitidos durante as operações entre regiões permanecem na rede da AWS e não atravessam a Internet pública. Criptografamos dados em trânsito entre as regiões da AWS.

A tabela a seguir descreve onde suas solicitações de inferência são processadas com base na região de origem da solicitação e no recurso usado.

Origem da solicitação	Todos os recursos, exceto a remediação de código	Remediação de código
Estados Unidos da América — Leste dos EUA (Norte da Virgínia) `us-east-1` —, Oeste dos EUA (Oregon) — `us-west-2`	Estados Unidos	Estados Unidos

Origem da solicitação	Todos os recursos, exceto a remediação de código	Remediação de código
União Europeia — Europa (Irlanda) `eu-west-1` —, Europa (Frankfurt) — `eu-central-1`	União Europeia	União Europeia
Austrália — Ásia-Pacífico (Sydney) — `ap-southeast-2`	Austrália	União Europeia
Japão — Ásia-Pacífico (Tóquio) — `ap-northeast-1`	Japão	União Europeia
América do Sul — América do Sul (São Paulo) — `sa-east-1`	Qualquer região comercial da AWS	Qualquer região comercial da AWS
Índia — Ásia-Pacífico (Mumbai) — `ap-south-1`	Qualquer região comercial da AWS	Qualquer região comercial da AWS
Sudeste Asiático — Ásia-Pacífico (Singapura) — `ap-southeast-1`	Qualquer região comercial da AWS	Qualquer região comercial da AWS

Cross-Region a inferência está sempre ativada e não pode ser excluída. Cross-Region a inferência não é afetada pelas políticas do cliente nas Políticas de Controle de Serviços (SCPs) ou na AWS Control Tower que restringem o conteúdo do cliente a regiões específicas. Para obter mais informações sobre como o AWS Security Agent protege seus dados durante o processamento entre regiões, consulte Processamento de Cross-Region dados.

Migração de ações e recursos do IAM

O AWS Security Agent é um agente de fronteira que protege proativamente seus aplicativos durante todo o ciclo de vida de desenvolvimento em todos os seus ambientes. Se você se inscreveu no AWS Security Agent antes de 9 de fevereiro de 2026, você será afetado pelas próximas mudanças em 9

de março de 2026 nos seus recursos atuais de instância de agente e nas ações do AWS Security Agent IAM. Em preparação para o lançamento do API/SDK suporte público, o recurso Agent Instance está sendo renomeado para Agent Space e ações específicas do IAM estão sendo renomeadas. Essas alterações afetarão todas as funções do IAM de Application ou Agent Instances que você tenha criado antes de 9 de março de 2026. Para evitar problemas de autenticação após 9 de março de 2026, você precisará seguir as etapas em Preparação para a migração.

> **Note**
>
> Se você criar novas instâncias do agente após 9 de fevereiro de 2026, a nova instância do agente será criada como um espaço do agente e nenhuma etapa de migração será necessária.

Mudanças planejadas

O AWS Security Agent está renomeando o recurso Agent Instance para Agent Space: `arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*` renomeado para. `arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*` Além disso, as seguintes ações do IAM estão sendo renomeadas:

- `securityagent:ListAgentInstances` renomeado para `securityagent:ListAgentSpaces`
- `securityagent:ListControls` renomeado para `securityagent:ListSecurityRequirements`
- `securityagent:BatchGetAgentInstances` renomeado para `securityagent:BatchGetAgentSpaces`
- `securityagent:BatchGetSecurityTestContentMetadata` renomeado para `securityagent:BatchGetPentestJobContentMetadata`
- `securityagent:BatchGetTasks` renomeado para `securityagent:BatchGetPentestJobTasks`
- `securityagent:CreateDocumentReview` renomeado para `securityagent:CreateDesignReview`
- `securityagent:GetDocumentReview` renomeado para `securityagent:GetDesignReview`
- `securityagent:GetDocumentReviewArtifact` renomeado para `securityagent:GetDesignReviewArtifact`

- `securityagent:ListDocumentReviews` renomeado para `securityagent:ListDesignReviews`
- `securityagent:ListDocumentReviewComments` renomeado para `securityagent:ListDesignReviewComments`
- `securityagent:ListTasks` renomeado para `securityagent:ListPentestJobTasks`
- `securityagent:StartPentestExecution` renomeado para `securityagent:StartPentestJob`
- `securityagent:StopPentestExecution` renomeado para `securityagent:StopPentestJob`
- `securityagent:DeleteDocumentReview` renomeado para `securityagent:DeleteDesignReview`

Preparação para a migração

Para evitar problemas após 9 de março de 2026 e continuar usando o AWS Security Agent antes de 9 de março de 2026, você precisará confiar nos recursos novos e antigos e nas ações do IAM em seu IAM roles/policies até 9 de março de 2026. As instruções abaixo fornecerão um guia para migrar para os novos formatos de recursos e ações:

1. Faça login na sua conta da AWS e navegue até o console do AWS Security Agent
2. No painel esquerdo, selecione Configurações e escolha a função em Função de serviço
3. No console do IAM para a função associada, selecione Adicionar permissões e Anexar políticas
4. Selecione AWSSecurityAgentWebAppPolicye escolha Adicionar permissões
   a. Nota importante: Verifique se você selecionou AWSSecurityAgentWebAppPolicycomo a nova política e não SecurityAgentWebAppAPIPolicy
5. Verifique se sua função do IAM agora tem ambas as políticas de permissões AWSSecurityAgentWebAppPolicye SecurityAgentWebAppAPIPolicyem
6. No mesmo console de funções do IAM, selecione Relações de confiança e, em seguida, Editar política de confiança
7. Atualize sua política de confiança para o seguinte formato, substituindo {{accountId}} pelo ID da sua conta da AWS

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:securityagent:us-east-1:{{accountId}}:application/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*"
          ]
        }
      }
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:SetContext",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}"
        },
        "ForAllValues:ArnEquals": {
          "sts:RequestContextProviders": "arn:aws:iam::aws:contextProvider/
IdentityCenter"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:securityagent:us-east-1:{{accountId}}:application/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*"
          ]
        }
      }
    }
  ]
```

```
}
```

8. Volte para o console do AWS Security Agent. No painel esquerdo, selecione Agent Spaces
9. Para cada Espaço de Agente que você tem com o teste de penetração ativado, execute as seguintes etapas
   a. Navegue até o Espaço do Agente e selecione Teste de penetração
   b. Em Acesso ao serviço, escolha a função em Nome da função do serviço
   c. No console do IAM para a função associada, selecione Relações de confiança e, em seguida, Editar política de confiança
   d. Atualize sua política de confiança para o seguinte formato, substituindo {{accountId}} pelo ID da sua conta da AWS

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*"
          ]
        }
      }
    }
  ]
}
```

Registro em log AWS Chamadas de API do Security Agent com AWS CloudTrail

AWS O Security Agent é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no AWS Security Agent. CloudTrail captura todas as chamadas de API para o AWS Security Agent como eventos. As chamadas capturadas incluem chamadas do console do AWS Security Agent e chamadas de código para as operações da API do AWS Security Agent. Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para o AWS Security Agent. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita ao AWS Security Agent, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para saber mais sobre isso CloudTrail, consulte o Guia AWS CloudTrail do usuário.

AWS Informações do Security Agent em CloudTrail

CloudTrail é ativado em sua AWS conta quando você cria a conta. Quando a atividade ocorre no AWS Security Agent, essa atividade é registrada em um CloudTrail evento junto com outros eventos AWS de serviço no histórico de eventos. Você pode visualizar, pesquisar e baixar eventos recentes em sua AWS conta. Para obter mais informações, consulte Visualização de eventos com histórico de CloudTrail eventos.

Para um registro contínuo dos eventos em sua AWS conta, incluindo eventos do AWS Security Agent, crie uma trilha. Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Por padrão, quando você cria uma trilha no console, a trilha se aplica a todas as AWS regiões. A trilha registra eventos de todas as regiões na AWS partição e entrega os arquivos de log ao bucket do Amazon S3 que você especificar. Além disso, você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para saber mais, consulte:

- Visão geral da criação de uma trilha
- CloudTrail serviços e integrações suportados
- Configurando notificações do Amazon SNS para CloudTrail
- Recebendo arquivos de CloudTrail log de várias regiões e Recebendo arquivos de CloudTrail log de várias contas

Todas as ações do AWS Security Agent são registradas por CloudTrail. Por exemplo, chamadas para as `ListFindings` ações `CreatePentestBatchGetPentestJobs`, e geram entradas nos arquivos de CloudTrail log.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar o seguinte:

- Se a solicitação foi feita com credenciais de usuário root ou de usuário do AWS Identity and Access Management (IAM).
- Se a solicitação foi feita com credenciais de segurança temporárias de um perfil ou de um usuário federado.
- Se a solicitação foi feita por outro AWS serviço.

Para obter mais informações, consulte Elemento userIdentity do CloudTrail .

Exemplo: AWS Entradas do arquivo de log do Security Agent

Noções básicas AWS Entradas do arquivo de log do Security Agent

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3 que você especificar. CloudTrail os arquivos de log contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `CreatePentest` ação:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/Alice",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "Alice"
    },
    "eventTime": "2025-01-15T10:30:00Z",
```

```
    "eventSource": "securityagent.amazonaws.com",
    "eventName": "CreatePentest",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "203.0.113.12",
    "userAgent": "aws-cli/2.13.0",
    "requestParameters": {
        "pentestName": "WebApp-Security-Test",
        "targetUrl": "https://example.com",
        "testScope": "OWASP-Top-10"
    },
    "responseElements": {
        "pentestId": "pt-1234567890abcdef0",
        "pentestArn": "arn:aws:securityagent:us-east-1:123456789012:pentest/
pt-1234567890abcdef0"
    },
    "requestID": "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
    "eventID": "12345678-1234-1234-1234-123456789012",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```


Service Quotas

O AWS Security Agent tem cotas que limitam o número de recursos que você pode criar e a taxa na qual você pode realizar operações. As cotas marcadas como ajustáveis podem ser aumentadas enviando uma solicitação por meio do AWS Support. Para obter mais informações, consulte Criar casos de suporte e gerenciamento de casos.

Cotas de operações

As cotas de operações limitam o uso mensal dos recursos de teste e análise de segurança para ajudar a gerenciar a capacidade do serviço.

Recurso	Escopo	Quota	Ajustável
Revisões de design	Por mês, por conta, por região	200	Sim

Recurso	Escopo	Quota	Ajustável
Revisões de código de relações públicas	Por mês, por conta, por região	1.000	Sim

Cotas de configuração

As cotas de configuração limitam o número de recursos e configurações que você pode definir em seu ambiente do AWS Security Agent.

Recurso	Escopo	Quota	Ajustável
Espaços para agentes	Por conta por região	100	Sim
Integrações	Por conta por região	20	Não
Recursos integrados por integração	Por integração	50	Não
Requisitos de segurança personalizados	Por conta por região	20	Não
Projetos Pentest	Por conta por região	1.000	Sim
Execuções simultâneas de pentest	Por conta por região	5	Sim
Projetos de revisão de código	Por conta por região	1.000	Sim
Execuções simultâneas de revisão de código	Por conta por região	5	Sim

Histórico do documento

A tabela a seguir descreve algumas das principais atualizações e novos recursos do Guia do usuário do AWS Security Agents.

Alteração	Descrição	Data

Conexões privadas (versão prévia)	Documentação adicionada para conexões privadas. Esse novo recurso permite que o AWS Security Agent se conecte a sistemas de controle de origem executados em redes privadas usando o Amazon VPC Lattice, sem expor seus sistemas à Internet pública.	16 de junho de 2026
Modelagem de ameaças (versão prévia)	Documentação adicionada para modelagem de ameaças. Esse novo recurso cria um modelo de ameaça para seu aplicativo a partir de documentos de design (documentos de escopo), código-fonte (fontes) ou ambos. Os documentos de escopo são documentos de design de recursos que definem o foco da análise, enquanto o código-fonte fornece contexto sobre seu sistema existente. Se você não fornecer documentos de escopo, o agente gera um modelo de ameaça apenas a partir do código-fonte. Cada execução produz uma visão geral do sistema e um conjunto de ameaças classificadas por categoria STRIDE com classificações de gravidade e recomendações.	8 de junho de 2026

Análise completa do código do repositório (pré-visualização)	Documentação adicionada para análise completa do código do repositório. Esse novo recurso realiza uma análise de segurança sensível ao contexto de toda a sua base de código e gera remediação de código para descobertas.	12 de maio de 2026
Disponibilidade atualizada da região para encontrar remediação	A disponibilidade da região para testes de penetração encontrando remediação no aplicativo web do AWS Security Agent foi atualizada.	16 de abril de 2026
Disponibilidade atualizada da região para permitir a busca de remediação	A disponibilidade da região para permitir o teste de penetração encontrando remediação no AWS Management Console foi atualizada.	16 de abril de 2026
Suporte de chave gerenciada pelo cliente	Documentação adicionada para suporte à chave gerenciada pelo cliente (CMK). Agora você pode especificar uma chave KMS gerenciada pelo cliente ao criar espaços de agentes e integrações para criptografar seus dados com chaves que você controla.	31 de março de 2026

AWS atualizações de políticas gerenciadas	Foi AWSSecurityAgentWebAppPolicy adicionada uma política gerenciada para os tipos novos TargetDomain e de DesignReviewFeedback recursos.	31 de março de 2026
AWS atualizações de políticas gerenciadas	Foi adicionada uma política AWSSecurityAgentWebAppPolicy gerenciada para o novo tipo de AgentSpace recurso e alterações no nome da ação do IAM.	9 de fevereiro de 2026
AWS atualizações de políticas gerenciadas	Atualizado SecurityAgentWebAppAPIPolicy para permitir que os clientes excluam avaliações de design.	28 de janeiro de 2026
AWS atualizações de políticas gerenciadas	Atualizado SecurityAgentWebAppAPIPolicy para permitir que os clientes iniciem a correção automática de código para descobertas de segurança.	20 de janeiro de 2026
AWS atualizações de políticas gerenciadas	Atualizado SecurityAgentWebAppAPIPolicy para permitir que os clientes visualizem imagens no console.	5 de dezembro de 2025
Lançamento inicial do AWS Security Agents	Documentação inicial para a inicialização do serviço	2 de dezembro de 2025

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.