aws

Guía del usuario de

Amazon Verified Permissions

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

Amazon Verified Permissions: Guía del usuario de

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

Las marcas registradas y la imagen comercial de Amazon no se pueden utilizar en ningún producto o servicio que no sea de Amazon de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

¿Qué es Amazon Verified Permissions? 1
Autorización en Verified Permissions 1
Lenguaje político de Cedar 2
Ventajas de Verified Permissions 2
Agilizar el desarrollo de las aplicaciones 2
Aplicaciones más seguras 2
End-user features 3
Servicios relacionados 3
Acceso a Verified Permissions 3
Precios de Verified Permissions 5
Cómo empezar con las tiendas de políticas 6
Requisitos previos 7
Regístrate para obtener un Cuenta de AWS 7
Paso 1: crear un PhotoFlash tienda de políticas 7
Paso 2: Crea una política 8
Paso 3: Probar un almacén de políticas 9
Paso 4: limpie los recursos 10
Diseño de un modelo de autorización 12
No hay un único modelo correcto 13
Devolución de errores 14
Centrarse en los recursos 14
Considere la posibilidad de tener varios arrendatarios 16
Comparación de los almacenes de políticas compartidos y los almacenes de políticas por inquilino 17
¿Cómo elegir 18
Almacenes de políticas 20
Crear almacenes de políticas 21
Crear un almacén de políticas con Rust 29
Almacenes de políticas vinculados a API 35
Funcionamiento 37
Consideraciones 38
Añadir ABAC 40
Pasar a la producción 41
Resolución de problemas 43

Eliminar almacenes de políticas 46
Alias de la tienda de políticas 49
Propiedades de los alias de los almacenes de políticas 49
Creación de alias de almacenes de políticas 53
Recuperando los alias de la tienda de políticas 54
Obtener un alias de almacén de políticas 54
Listar los alias de las tiendas de políticas 55
Eliminar los alias de la tienda de políticas 56
Eliminación gradual de un alias de almacén de políticas 57
Eliminar de forma difícil un alias de almacén de políticas 57
Uso de los alias de las tiendas de políticas 58
Uso de los alias del almacén de políticas en Operations 59
Uso de los alias del almacén de políticas en Across Regiones de AWS 59
Los alias del almacén de políticas no son un mecanismo de control del tráfico 60
Realizar cambios en el almacén de políticas sin tiempo de inactividad 62
Control del acceso 63
permisos verificados: CreatePolicyStoreAlias 63
permisos verificados: GetPolicyStoreAlias 64
permisos verificados: ListPolicyStoreAliases 64
permisos verificados: DeletePolicyStoreAlias 65
Limitar los permisos del alias del almacén de políticas 65
Esquema del almacén de políticas 68
Edición del esquema 70
Modo de validación de políticas 73
Políticas 75
Creación de políticas estáticas 76
Edición de políticas estáticas 79
..... 82
Evalúe el contexto del ejemplo 84
Políticas de pruebas 90
Tamaño de la política por recurso 92
Cómo se calculan los tamaños de las pólizas 93
Mantenerse dentro de la cuota 93
Template-linked ejemplo de tamaño de política 94
Ejemplos de políticas 97
Usa la notación entre corchetes para hacer referencia a los atributos del token 97

Usa la notación de puntos para hacer referencia a los atributos 98
Refleja Amazon Cognito Atributos del token de identificación 98
Refleja los atributos del token de ID de OIDC 99
Refleja Amazon Cognito atributos del token de acceso 99
Refleja los atributos del token de acceso de OIDC 100
Plantillas de políticas y políticas vinculadas a plantillas 101
Crear plantillas de política 102
Crear políticas vinculadas a plantillas 104
Editar plantillas de política 106
Ejemplo de políticas vinculadas a plantillas 108
PhotoFlash ejemplos 108
DigitalPetStore ejemplos 110
TinyToDo ejemplos 110
Fuentes de identidad 112
Elegir el proveedor de identidad correcto 113
Trabajar con fuentes Amazon Cognito de identidad 113
Crear fuentes de identidad 116
Editar fuentes de identidad 120
Asignación de tokens al esquema 122
Validación de clientes y audiencias 133
Trabajar con fuentes de identidad del OIDC 136
Crear fuentes de identidad 138
Editar fuentes de identidad 141
Asignación de los tokens al esquema 143
Validación de clientes y audiencias 150
Integraciones 155
Uso de Express 155
Requisitos previos 156
Configuración de la integración 156
Configuración de la autorización 157
Implementación del middleware de autorización 160
Probando la integración 161
Resolución de problemas 161
Siguientes pasos 161
Autoriza las solicitudes 162
Operaciones de la API 163

Pruebe el modelo ... 164
Integración con aplicaciones ... 166
Cómo realizar solicitudes a la API ... 169
Puntos finales de permisos verificados ... 169
Parámetros de solicitud ... 169
Identificadores de solicitud ... 169
Consulta la autenticación de la API ... 169
Bibliotecas disponibles ... 169
Realizar solicitudes a la API mediante el método POST ... 170
Seguridad ... 173
Protección de datos ... 173
Cifrado de datos ... 175
Claves administradas por el cliente ... 175
Identity and Access Management ... 195
Público ... 196
Autenticación con identidades ... 196
Administración del acceso con políticas ... 199
Cómo funciona Amazon Verified Permissions con IAM ... 201
IAM políticas de permisos verificados ... 206
Ejemplos de políticas basadas en identidades ... 209
AWS políticas gestionadas ... 213
Resolución de problemas ... 216
Validación de conformidad ... 218
Resiliencia ... 219
Supervisión ... 220
CloudTrail registros ... 220
La información de permisos verificados se encuentra en CloudTrail ... 221
Descripción de las entradas del archivo de registro de Verified Permissions ... 222
Trabajando con AWS CloudFormation ... 244
Permisos y plantillas verificados CloudFormation ... 244
AWS CDK construye ... 245
Más información sobre CloudFormation ... 245
Usando AWS PrivateLink ... 246
Consideraciones ... 246
Crear un punto de conexión de interfaz ... 246
Creación de una política de punto de conexión ... 247

Cuotas 249
Cuotas de recursos 249
Cuotas para jerarquías 251
Cuotas de operaciones por segundo 252
Términos y conceptos 257
Modelo de autorización 258
Solicitud de autorización 258
Respuesta de autorización 258
Políticas consideradas 258
Datos de contexto 259
Políticas determinantes 259
Datos de la entidad 259
Permisos, autorizaciones y entidades principales 259
Aplicación de políticas 259
Almacén de políticas 260
Alias del almacén de políticas 260
Nombre de la política 260
Nombre de la plantilla de política 261
Políticas satisfechas 261
Diferencias con Cedar 261
Compatibilidad con las plantillas de política 261
Compatibilidad con esquemas 262
Definición de grupos de acción 262
Formato de entidades 262
Límites de longitud y tamaño 267
Preguntas frecuentes sobre Cedar v4 269
¿Por qué recibo un mensaje de error que dice que Cedar 2 ya no es compatible cuando llamo a Amazon Verified Permissions? 269
¿Por qué algunas políticas, plantillas de políticas y esquemas no son compatibles con Cedar 4? 270
¿Cómo puedo saber si mi almacén de pólizas usa Cedar 2 o Cedar 4? 270
¿Cómo puedo actualizar a Cedar 4? 272
¿Puedo cambiar mi almacén de pólizas de Cedar 4 a Cedar 2? 272
¿Por qué recibo un mensaje de error que indica que mi almacén de políticas está configurado para Cedar 2? 272
¿Cómo hago que mi esquema sea compatible con Cedar 4? 273

¿Cómo puedo hacer que mis políticas y plantillas sean compatibles con Cedar 4? 274
Historial de revisión 276
..... cclxxviii

¿Qué es Amazon Verified Permissions?

Amazon Verified Permissions es un servicio de autorización y administración de permisos escalable y detallado para aplicaciones personalizadas diseñado para usted. Verified Permissions permite a sus desarrolladores crear aplicaciones seguras con mayor rapidez al externalizar la autorización y centralizar la gestión y la administración de las políticas. Verified Permissions utiliza el lenguaje de políticas de Cedar para definir permisos detallados a fin de proteger los recursos de su aplicación.

Para obtener orientación y ejemplos sobre cómo configurar un punto de decisión política (PDP) mediante permisos verificados, consulte Implementación de un PDP mediante los permisos verificados de Amazon en AWS Orientación prescriptiva.

Temas

- Autorización en Verified Permissions
- Lenguaje político de Cedar
- Ventajas de Verified Permissions
- Servicios relacionados
- Acceso a Verified Permissions
- Precios de Verified Permissions

Autorización en Verified Permissions

Verified Permissions proporciona autorización verificando si un principal está autorizado a realizar una acción en un recurso en un contexto determinado de su aplicación. Los permisos verificados suponen que el principal ha sido previamente identificado y autenticado por otros medios, como el uso de protocolos como OpenID Connect, un proveedor hospedado como Amazon Cognito, u otra solución de autenticación. Los permisos verificados son independientes del lugar donde se administra el principal y de la forma en que se autenticó.

Los permisos verificados son un servicio que permite a los clientes crear, mantener y probar políticas en Consola de administración de AWS, mediante programación, las API de permisos verificados o mediante la infraestructura como soluciones de código, como CloudFormation. Los permisos se expresan utilizando el lenguaje de políticas de Cedar. La aplicación cliente llama a las API de autorización para evaluar las políticas de Cedar almacenadas con el servicio y decidir si se permite o no una acción en materia de acceso.

Lenguaje político de Cedar

Las políticas de autorización de Verified Permissions se redactan utilizando el lenguaje de políticas de Cedar. Cedar es un lenguaje de código abierto para redactar políticas de autorización y tomar decisiones de autorización basadas en esas políticas. Al crear una aplicación, debe asegurarse de que solo los directores autorizados, los usuarios humanos o las máquinas, puedan acceder a la aplicación y solo puedan hacer lo que están autorizados a hacer. Con Cedar, puede desvincular la lógica empresarial de la lógica de autorización. En el código de la aplicación, preceda las solicitudes realizadas a sus operaciones con una llamada al motor de autorización de Cedar con esta pregunta: "¿Está autorizada esta solicitud?". Después, la aplicación puede realizar la operación solicitada si la decisión es "permitir" o devolver un mensaje de error si la decisión es "denegar".

Verified Permissions utiliza actualmente la versión 4.12 de Cedar.

Para obtener más información sobre Cedar, consulte lo siguiente:

- Guía de referencia sobre el lenguaje de políticas de Cedar
- Cedro GitHub repositorio

Ventajas de Verified Permissions

Agilizar el desarrollo de las aplicaciones

Agilice el desarrollo de las aplicaciones separando la autorización de la lógica empresarial.

Verified Permissions proporciona integraciones con marcos de desarrollo populares, lo que facilita la implementación de la autorización en sus aplicaciones con cambios mínimos en el código. Estas integraciones le permiten centrarse en la lógica empresarial principal, mientras que Verified Permissions se encarga de las decisiones de autorización.

- Express.js — Una integración basada en middleware que le permite proteger los puntos finales de la API en sus aplicaciones Express sin modificar los controladores de rutas existentes. Para obtener más información, consulte the section called "Uso de Express".

Aplicaciones más seguras

Verified Permissions permite a los desarrolladores crear aplicaciones más seguras.

End-user features

Verified Permissions permite ofrecer características de usuario final más completas para la administración de permisos.

Servicios relacionados

- Amazon Cognito: es una plataforma de identidad para aplicaciones web y móviles. Es un directorio de usuarios, un servidor de autenticación y un servicio de autorización para los tokens y credenciales de AWS de acceso de OAuth 2.0. Al crear un almacén de políticas, tiene la opción de crear sus principales y grupos a partir de un Amazon Cognito grupo de usuarios. Para obtener más información, consulte la Guía para desarrolladores de Amazon Cognito.
- Amazon API Gateway — Amazon API Gateway es un AWS servicio para crear, publicar, mantener, monitorear y proteger REST, HTTP y WebSocket API a cualquier escala. Cuando creas un almacén de políticas, tienes la opción de crear tus acciones y recursos a partir de una API en API Gateway. Para obtener más información sobre API Gateway, consulte el API Gateway Guía para desarrolladores.
- AWS IAM Identity Center: con IAM Identity Center, puede gestionar la seguridad de inicio de sesión de las identidades de sus empleados, también conocidos como usuarios de los empleados. IAM Identity Center ofrece un lugar donde puede crear o conectar usuarios de la fuerza laboral y gestionar de forma centralizada su acceso en todos sus Cuentas de AWS y aplicaciones. Para obtener más información, consulte la Guía del usuario de AWS IAM Identity Center.

Acceso a Verified Permissions

Puede trabajar con Amazon Verified Permissions de cualquiera de las siguientes formas:

Consola de administración de AWS

La consola es una interfaz basada en navegador para administrar Verified Permissions y los recursos de AWS . Para obtener más información acerca de cómo acceder a Verified Permissions mediante la consola, consulte Cómo iniciar sesión en AWS en la Guía del usuario de AWS Sign-In .

- Consola de permisos verificados de Amazon

AWS Herramientas de línea de comandos

Puede utilizar el AWS herramientas de línea de comandos para emitir comandos en la línea de comandos del sistema para ejecutar los permisos verificados y AWS tareas. El uso de la línea de comandos puede ser más rápido y cómodo que la consola. Las herramientas de línea de comandos también son útiles si desea crear scripts que realicen tareas de AWS .

AWS proporciona dos conjuntos de herramientas de línea de comandos: AWS Command Line Interface (AWS CLI) y el AWS Tools for Windows PowerShell. Para obtener información sobre la instalación y el uso del AWS CLI, consulte el AWS Command Line Interface Guía del usuario. Para obtener información sobre la instalación y el uso de las Herramientas para Windows PowerShell, consulte el Herramientas de AWS para PowerShell Guía del usuario.

- permisos verificados en el AWS CLI Referencia de comandos
- Permisos verificados de Amazon en AWS Tools for Windows PowerShell

AWS SDKs

AWS proporciona SDK (kits de desarrollo de software) que consisten en bibliotecas y código de muestra para varios lenguajes de programación y plataformas (Java, Python, Ruby, .NET, iOS, Android, etc.). Los SDK proporcionan una forma cómoda de crear acceso mediante programación a Verified Permissions y AWS. Por ejemplo, los SDK se encargan de tareas como firmar solicitudes criptográficamente, gestionar los errores y reintentar las solicitudes de forma automática.

Para obtener más información y descargar AWS Los SDK, consulte Herramientas para Amazon Web Services.

Los siguientes son enlaces a la documentación de los recursos de permisos verificados en varios AWS SDK.

- AWS SDK para .NET
- AWS SDK para C++
- AWS SDK para Go
- AWS SDK para Java
- AWS SDK para JavaScript
- AWS SDK para PHP
- AWS SDK para Python (Boto)
- AWS SDK para Ruby

- AWS SDK para Rust

AWS Construcciones de CDK

El Kit de desarrollo de la nube de AWS (AWS CDK) es un marco de desarrollo de software de código abierto para definir la infraestructura de nube en código y aprovisionarla a través de CloudFormation. Las construcciones, o componentes de nube reutilizables, se pueden utilizar para crear CloudFormation plantillas. Luego, estas plantillas se pueden usar para implementar su infraestructura de nube.

Para obtener más información y descargarlas AWS CDK, consulte AWS Kit de desarrollo en la nube.

Los siguientes son enlaces a la documentación de los permisos verificados AWS CDK recursos, como construcciones.

- Permisos verificados de Amazon: L2 CDK Construct

API de Verified Permissions

Puede acceder a los permisos verificados y AWS mediante programación mediante la API de permisos verificados, que te permite emitir solicitudes HTTPS directamente al servicio. Cuando use la API, debe incluir código para firmar digitalmente las solicitudes utilizando sus credenciales.

- Guía de referencia de la API de permisos verificados de Amazon

Precios de Verified Permissions

Verified Permissions ofrece precios escalonados en función de la cantidad de solicitudes de autorización que realicen al mes sus solicitudes a Verified Permissions. También hay precios para las acciones de administración de políticas en función de la cantidad de solicitudes mensuales de la API de políticas cURL (URL del cliente) que realicen sus aplicaciones a Verified Permissions.

Para obtener una lista completa de los cargos y precios de Verified Permissions, consulte Precios de Amazon Verified Permissions.

Para ver su factura, vaya al Panel de Billing and Cost Management en la consola de Administración de facturación y costos de AWS. La factura contiene vínculos a informes de uso que ofrecen detalles sobre la cuenta. Para obtener más información sobre Cuenta de AWS facturación, consulta el AWS Billing Guía del usuario.

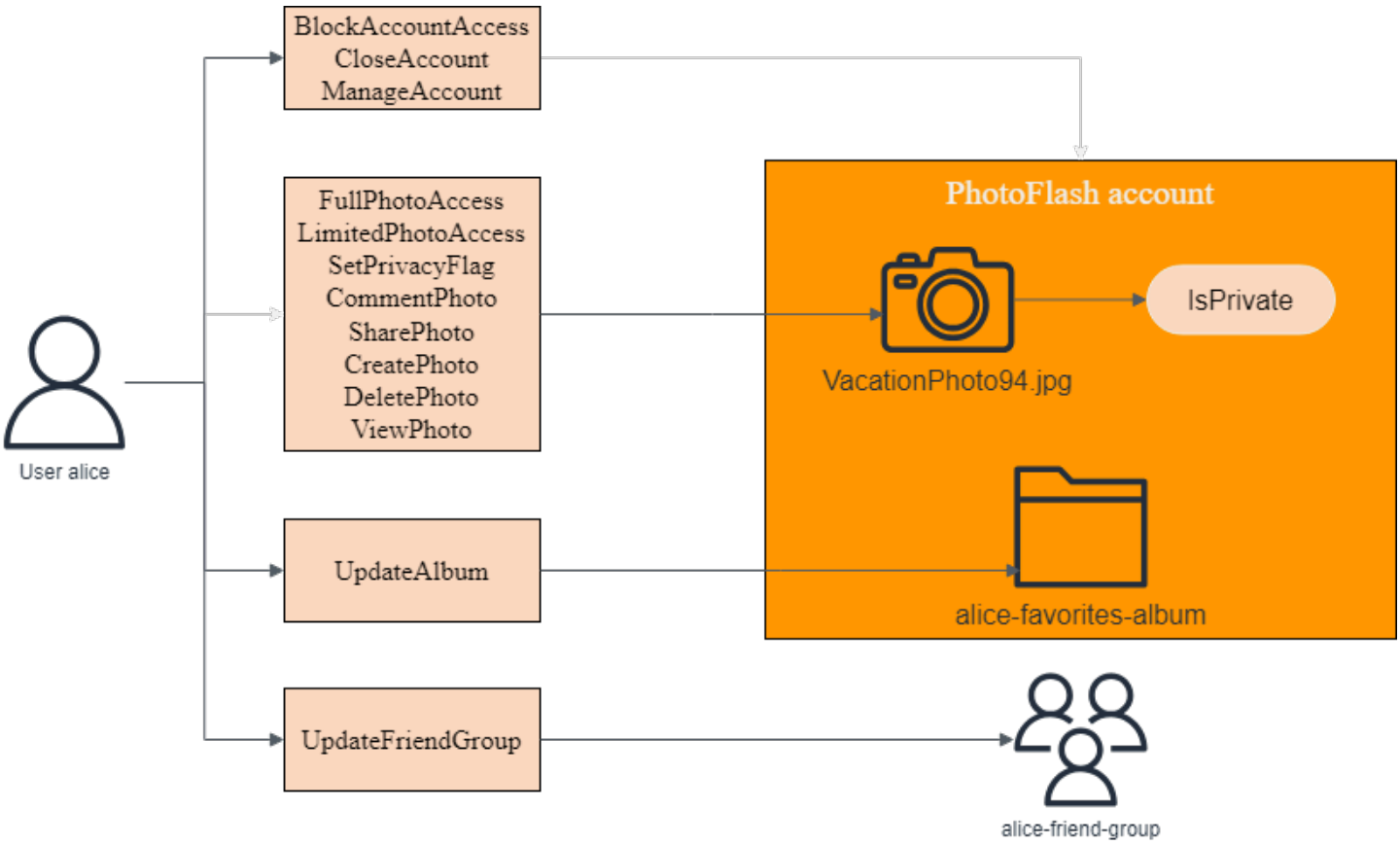
Si tiene preguntas sobre AWS facturación, cuentas y eventos, contacto Soporte.

Crea tu primera tienda de políticas de permisos verificados de Amazon

Para este tutorial, supongamos que eres el desarrollador de una aplicación para compartir fotos y estás buscando una forma de controlar las acciones que pueden realizar los usuarios de la aplicación. Quieres controlar quién puede añadir, eliminar o ver fotos y álbumes de fotos. También quieres controlar qué acciones puede realizar un usuario en su cuenta. ¿Pueden administrar su cuenta? ¿Qué tal la cuenta de un amigo? Para controlar estas acciones, debes crear políticas que permitan o prohíban estas acciones en función de la identidad del usuario. Ofertas de permisos verificados tiendas de políticas, o contenedores, para albergar estas políticas.

En este tutorial veremos cómo crear un ejemplo de almacén de políticas con la consola de permisos verificados de Amazon. La consola ofrece algunos ejemplos de opciones de almacenamiento de políticas y vamos a crear un PhotoFlash tienda de políticas. Este almacén de políticas permite directores, como los usuarios, para realizar acciones, como compartir, en recursos, como fotos o álbumes.

El siguiente diagrama ilustra las relaciones entre un director, `User::alice`, y las medidas que puede tomar con respecto a varios recursos, a saber, su PhotoFlash cuenta, la `VactionPhoto94.jpg` archivo, el álbum de fotos `alice-favorites-album`, y el grupo de usuarios `alice-friend-group`.



Ahora que tiene una comprensión de las PhotoFlash almacén de políticas, vamos a crear el almacén de políticas y explorarlo.

Requisitos previos

Regístrate para obtener un Cuenta de AWS

Para empezar con AWS, necesitas un Cuenta de AWS. Hay dos formas de inscribirse en AWS. Si quieres tener un control total sobre la configuración de la cuenta y dónde se almacena tu contenido, o si tienes cargas de trabajo reguladas, como la HIPAA o el FedRAMP, utiliza Inscríbase en AWS (avanzado). Si es un usuario nuevo y desea utilizar un nombre de usuario que ya posee para iniciar sesión AWS y crea un AWS entorno con valores predeterminados preconfigurados en Select Regiones de AWS, utilice Inscríbase para AWS.

Paso 1: crear un PhotoFlash tienda de políticas

En el siguiente procedimiento, creará un PhotoFlash almacén de políticas mediante AWS consola.

Para crear un PhotoFlash almacén de políticas

1. En el Consola de permisos verificados, elige Crear un nuevo almacén de políticas.
2. Para Opciones de inicio, elige Comience desde un almacén de políticas de muestra.
3. Para Ejemplo de proyecto, elige PhotoFlash.
4. Seleccione Crear almacén de políticas.

Cuando veas el mensaje «Almacén de políticas creado y configurado», elige Ir a la descripción general para explorar tu tienda de pólizas.

Paso 2: Crea una política

Al crear el almacén de políticas, se creó una política predeterminada que permite a los usuarios tener un control total sobre sus propias cuentas. Esta es una política útil, pero para nuestros propósitos, vamos a crear una política más restrictiva para explorar los matices de los permisos verificados. Si recuerdas el diagrama que vimos anteriormente en el tutorial, teníamos un director, `User::alice`, quién podía realizar una acción, `UpdateAlbum`, en un recurso, `alice-favorites-album`. Añadamos la política que permitirá a Alice, y solo a Alice, gestionar este álbum.

Creación de una política

1. En el Consola de permisos verificados, elige el almacén de políticas que creaste en el paso 1.
2. En la barra de navegación, elija Políticas.
3. Seleccione Crear política y, a continuación, elija Crear política estática.
4. Para Efecto de política, elige Permiso.
5. Para Alcance del director, elige Principal específico, entonces para Especifique el tipo de entidad, elija PhotoFlash::Usuario, y para Especifique el identificador de entidad, introduzca **`alice`**.
6. Para Alcance de los recursos, elige Recurso específico, luego para Especifique el tipo de entidad, elija PhotoFlash::Álbum, y para Especifique el identificador de entidad, introduzca **`alice-favorites-album`**.
7. Para Alcance de las acciones, elige Conjunto específico de acciones, luego para Acción (s) a las que debería aplicarse esta política, seleccione UpdateAlbum.
8. Elija Siguiente.

9. Bajo Detalles, para Descripción de la política: opcional entrar **`Policy allowing alice to update alice-favorites-album.`**.
10. Elija Create Policy

Ahora que has creado una política, puedes probarla en la consola de permisos verificados.

Paso 3: Probar un almacén de políticas

Después de crear su almacén de políticas y su política, puede probarlas ejecutando una simulación solicitud de autorización utilizando el banco de pruebas de permisos verificados.

Para probar las políticas del almacén de políticas

1. Abra el Consola de permisos verificados. Elige tu tienda de pólizas.
2. En el panel de navegación de la izquierda, seleccione Banco de pruebas.
3. Elija Modo visual.
4. Para Director, haga lo siguiente:
    a. Para El director está tomando medidas elegir PhotoFlash::Usuario y para Especifique el identificador de la entidad, introduzca **`alice`**.
    b. Bajo Atributos, para Cuenta: Entidad, asegúrese de que PhotoFlash::Cuenta se selecciona la entidad, y para Especifique el identificador de la entidad, introduzca **`alice-account`**.
5. Bajo Recurso, para Recurso sobre el que actúa el director, elige el PhotoFlash::Álbum tipo de recurso y para Especifique el identificador de la entidad, introduzca **`alice-favorites-album`**.
6. Para Acción, elige PhotoFlash::Acción::»UpdateAlbum» de la lista de acciones válidas.
7. En la parte superior de la página, elige Ejecuta la solicitud de autorización para simular la solicitud de autorización de las políticas de Cedar en el almacén de políticas de muestra. El banco de pruebas debería mostrar Decisión: Permitir indicando que nuestra política funciona según lo esperado.

La siguiente tabla proporciona valores adicionales para la entidad principal, el recurso y la acción que puede probar con el banco de pruebas de Verified Permissions. La tabla incluye la decisión sobre la solicitud de autorización basada en las políticas estáticas incluidas en el PhotoFlash ejemplo del almacén de políticas y la política que creó en el paso 2.

Valor de entidad principal	Valor Account: Entity de entidad principal	Valor de recurso	Valor de elemento principal del recurso	Action	Decisión de autorización
PhotoFlash::Usuario | bob	PhotoFlash::Cuenta | alice-account	PhotoFlash::Álbum | alice-favorito-album	N/A	PhotoFlash::Acción::»UpdateAlbum»	Denegar
PhotoFlash::Usuario | alice	PhotoFlash::Cuenta | alice-account	PhotoFlash::Foto | photo.jpeg	PhotoFlash::Cuenta | bob-account	PhotoFlash::Acción::»ViewPhoto»	Denegar
PhotoFlash::Usuario | alice	PhotoFlash::Cuenta | alice-account	PhotoFlash::Foto | photo.jpeg	PhotoFlash::Cuenta | alice-account	PhotoFlash::Acción::»ViewPhoto»	Permitir
PhotoFlash::Usuario | alice	PhotoFlash::Cuenta | alice-account	PhotoFlash::Foto | bob-photo.jpeg	PhotoFlash::Álbum | Bob-Vacation-Album	PhotoFlash::Acción::»DeletePhoto»	Denegar

Paso 4: limpie los recursos

Cuando hayas terminado de explorar tu almacén de políticas, elimínalo.

Para eliminar un almacén de políticas

1. En el Consola de permisos verificados, elige el almacén de políticas que creaste en el paso 1.
2. En la barra de navegación, elija Ajustes.
3. Bajo Eliminar almacén de políticas, elija Eliminar este almacén de políticas.

4. En el ¿Eliminar este almacén de políticas? cuadro de diálogo, introduzca eliminary, a continuación, elija Eliminar.

Mejores prácticas para diseñar un modelo de autorización

Mientras realiza los preparativos para utilizar el servicio Amazon Verified Permissions en una aplicación de software, puede resultar difícil pasar inmediatamente a redactar instrucciones de política como primer paso. Sería parecido a comenzar a desarrollar otras partes de una aplicación escribiendo instrucciones SQL o especificaciones de API antes de tener claro qué debe hacer la aplicación. En su lugar, deberías empezar con una experiencia de usuario. Después, partiendo de esa experiencia vaya hacia atrás para desarrollar una estrategia de implementación.

A medida que vaya realizando este trabajo, se hará preguntas como las siguientes:

- ¿Cuáles son mis recursos? ¿Cómo están organizados? Por ejemplo, ¿los archivos se encuentran dentro de una carpeta?
- ¿La organización de los recursos desempeña un papel en el modelo de permisos?
- ¿Qué acciones pueden realizar las entidades principales en cada recurso?
- ¿Cómo adquieren esos permisos las entidades principales?
- ¿Desea que sus usuarios finales elijan entre permisos predefinidos, como «Administrador», «Operador» u «ReadOnly», o deberían crear declaraciones de políticas ad hoc? ¿O ambos?
- ¿Los roles son globales o tienen un ámbito de aplicación? Por ejemplo, ¿un «operador» está limitado a un solo inquilino o por «operador» se entiende el operador de toda la aplicación?
- ¿Qué tipos de consultas son necesarios para ofrecer la experiencia de usuario? Por ejemplo, ¿necesita enumerar todos los recursos a los que puede acceder una entidad principal para mostrar la página de inicio de ese usuario?
- ¿Pueden los usuarios quedarse sin acceso a sus propios recursos por accidente? ¿Es necesario evitarlo?

El resultado final de este ejercicio se denomina modelo de autorización; define las entidades principales, los recursos, las acciones y la forma en que se interrelacionan entre sí. La elaboración de este modelo no requiere un conocimiento exclusivo de Cedar o del servicio Verified Permissions. Por el contrario, se trata ante todo de un ejercicio de diseño de la experiencia de usuario, como cualquier otro, y puede manifestarse en artefactos como maquetas de interfaces, diagramas lógicos y una descripción general de cómo los permisos influyen en lo que los usuarios pueden hacer en el producto. Cedar está diseñado para ser lo suficientemente flexible como para satisfacer las necesidades de los clientes en un modelo, en lugar de forzar al modelo a flexibilizarse de forma

poco natural para adaptarse a una implementación de Cedar. Por eso, tener una idea clara de la experiencia de usuario que se desea es la mejor manera de llegar a un modelo óptimo.

Para ayudar a responder a las preguntas y llegar a un modelo óptimo, haga lo siguiente:

- Revise los patrones de diseño de Cedar en la Guía de referencia sobre el lenguaje normativo de Cedar.
- Tenga en cuenta las mejores prácticas de la Guía de referencia sobre el lenguaje de las políticas de Cedar.
- Tenga en cuenta las mejores prácticas incluidas en esta página.

Prácticas recomendadas

- No existe un modelo canónico “correcto”
- Devuelve 403 errores prohibidos en lugar de 404 errores no encontrados
- Céntrese en sus recursos más allá de las operaciones de la API
- Consideraciones sobre la multitenencia

No existe un modelo canónico “correcto”

Cuando se diseña un modelo de autorización, no hay una respuesta única y única que sea correcta. Diferentes aplicaciones pueden utilizar de manera efectiva diferentes modelos de autorización para conceptos similares, y es algo que está bien. Por ejemplo, fíjese en la representación del sistema de archivos de un ordenador. Al crear un archivo en un sistema operativo similar a Unix, no hereda automáticamente los permisos de la carpeta principal. Por el contrario, en otros muchos sistemas operativos y en la mayoría de los servicios de intercambio de archivos en línea, los archivos heredan los permisos de su carpeta principal. Ambas opciones son válidas en función de las circunstancias para las que se esté optimizando la aplicación.

La corrección de una solución de autorización no es absoluta, pero debe considerarse en términos de cómo ofrece la experiencia que desean sus clientes y de si protege sus recursos de la manera que esperan. Si su modelo de autorización cumple con estos requisitos, entonces es correcto.

Por eso, empezar el diseño con la experiencia de usuario deseada es el requisito previo más útil para crear un modelo de autorización eficaz.

Devuelve 403 errores prohibidos en lugar de 404 errores no encontrados

Es mejor devolver el error 403 Forbidden a las solicitudes que incluyan una entidad, especialmente un recurso, que no corresponda a ninguna política en lugar de un error 404 No encontrado. Esto proporciona el nivel más alto de seguridad, ya que no estás revelando si una entidad existe o no, sino simplemente que la solicitud no cumplía las condiciones de política de ninguna política del almacén de políticas.

Céntrese en sus recursos más allá de las operaciones de la API

En la mayoría de las aplicaciones, los permisos se modelan en función de los recursos compatibles. Por ejemplo, una aplicación para compartir archivos puede representar los permisos como acciones que se pueden realizar en un archivo o una carpeta. Se trata de un modelo ilustrativo y sencillo que abstrae la implementación subyacente y las operaciones de la API de backend.

Por el contrario, otros tipos de aplicaciones, especialmente los servicios web, suelen diseñar los permisos en función de las propias operaciones de la API. Por ejemplo, si un servicio web proporciona una API llamada `createThing()`, el modelo de autorización puede definir el permiso correspondiente o uno de `action` en Cedar denominado `createThing`. Esto funciona en muchas situaciones y hace que sea más fácil entender los permisos. Para invocar la operación `createThing`, necesita el permiso de acción `createThing`. Parece sencillo, ¿verdad?

Descubrirás que el proceso de inicio de la consola de permisos verificados incluye la opción de crear tus recursos y acciones directamente desde una API. Se trata de una base útil: un mapeo directo entre el almacén de políticas y la API a la que se autoriza.

Sin embargo, a medida que vaya desarrollando su modelo, es posible que este enfoque centrado en las API no sea adecuado para las aplicaciones con modelos de autorización muy detallados, ya que no APIs son más que un indicador de lo que sus clientes realmente están intentando proteger: los datos y los recursos subyacentes. Si varios APIs controlan el acceso a los mismos recursos, puede resultar difícil para los administradores razonar sobre las rutas hacia esos recursos y gestionar el acceso en consecuencia.

Por ejemplo, pensemos en un directorio de usuarios que contenga los miembros de una organización. Los usuarios se pueden organizar en grupos y uno de los objetivos de seguridad es impedir que personas no autorizadas descubran la pertenencia a esos grupos. El servicio que administra el directorio de estos usuarios proporciona dos operaciones de API:

- `listMembersOfGroup`
- `listGroupMembershipsForUser`

Los clientes pueden usar cualquiera de estas operaciones para descubrir la pertenencia a un grupo. Por lo tanto, el administrador de los permisos debe acordarse de coordinar el acceso a ambas operaciones. Esto se complica aún más si más adelante decide añadir una nueva operación de la API para abordar casos de uso adicionales, como los siguientes.

- `isUserInGroups` (una nueva API para comprobar rápidamente si un usuario pertenece a uno o más grupos)

Desde el punto de vista de la seguridad, esta API abre una tercera vía para descubrir la pertenencia a grupos, lo cual altera los permisos cuidadosamente diseñados del administrador.

Le recomendamos que se centre en los datos y recursos subyacentes y en sus operaciones de asociación. Al aplicar este enfoque al ejemplo de pertenencia a un grupo, se obtendría un permiso abstracto, como `viewGroupMembership`, el cual deben consultar cada una de las tres operaciones de la API.

Nombre de API	Permisos
`listMembersOfGroup`	requiere el permiso `viewGroupMembership` del grupo
`listGroupMembershipsForUser`	requiere el permiso `viewGroupMembership` del usuario
`isUserInGroups`	requiere el permiso `viewGroupMembership` del usuario

Al definir este permiso, el administrador puede controlar el acceso al descubrimiento de la pertenencia a un grupo, ahora y siempre. Como desventaja, cada operación de la API ahora debe documentar los posibles permisos que necesite, y el administrador debe consultar esta documentación al crear los permisos. Pero puede ser una desventaja válida cuando sea necesaria para cumplir tus requisitos de seguridad.

Consideraciones sobre la multitenencia

Es posible que desee desarrollar aplicaciones para que las utilicen varios clientes (empresas que consumen su aplicación o inquilinos) e integrarlas con Amazon Verified Permissions. Antes de desarrollar su modelo de autorización, desarrolle una estrategia multiusuario. Puede administrar las políticas de sus clientes en un almacén de políticas compartido o asignar a cada uno un almacén de políticas por inquilino. Para obtener más información, consulte Consideraciones de diseño para varios inquilinos de Amazon Verified Permissions en la Guía AWS prescriptiva.

1. Un almacén de políticas compartido

   Todos los inquilinos comparten un único almacén de pólizas. La aplicación envía todas las solicitudes de autorización al almacén de políticas compartido.
2. Almacén de políticas por inquilino

   Cada inquilino tiene un almacén de pólizas dedicado. La aplicación consultará diferentes almacenes de pólizas para tomar una decisión de autorización, en función del inquilino que presente la solicitud.

Ninguna de estas estrategias tendrá un gran impacto en su AWS factura. Entonces, ¿cómo debería diseñar su enfoque? Las siguientes son condiciones comunes que podrían contribuir a su estrategia de autorización de arrendamiento múltiple con permisos verificados.

Políticas de inquilinos: aislamiento

El aislamiento de las políticas de cada inquilino de las demás es importante para proteger los datos del inquilino. Cuando cada inquilino tiene su propio almacén de políticas, cada uno tiene su propio conjunto aislado de políticas.

Flujo de autorización

Puede identificar a un inquilino que realiza una solicitud de autorización con un ID de almacén de políticas en la solicitud, si utiliza almacenes de políticas por inquilino. Con un almacén de políticas compartido, todas las solicitudes utilizan el mismo ID de almacén de políticas.

Administración de plantillas y esquemas

Cuando su aplicación tiene varios almacenes de políticas, sus plantillas de políticas y un esquema de almacén de políticas añaden un nivel de sobrecarga de diseño y mantenimiento a cada almacén de políticas.

Administración de políticas globales

Es posible que desee aplicar algunas políticas globales a todos los inquilinos. El nivel de gastos generales de administración de las políticas globales varía según el modelo de almacén de políticas compartido y el modelo por inquilino.

Inquilino abandona el embarque

Algunos inquilinos aportarán elementos a su esquema y políticas que sean específicos para su caso. Cuando un inquilino ya no está activo en su organización y usted desea eliminar sus datos, el nivel de esfuerzo varía en función de su nivel de aislamiento respecto a los demás inquilinos.

Cuotas de recursos de servicio

Verified Permissions tiene cuotas de recursos y tasas de solicitudes que pueden influir en tu decisión de tener varios arrendatarios. Para obtener más información sobre las cuotas, consulte Cuotas de recursos.

Comparación de los almacenes de políticas compartidos y los almacenes de políticas por inquilino

Cada consideración requiere su propio nivel de dedicación de tiempo y recursos en los modelos de almacenes de políticas compartidos y por inquilino.

Consideración	Nivel de esfuerzo en un almacén de políticas compartido	Nivel de esfuerzo en los almacenes de pólizas por inquilino
Aislamiento de políticas de inquilinos	Medio.Debe incluir los identificadores de los inquilinos en las políticas y solicitudes de autorización.	Bajo. El aislamiento es el comportamiento predeterminado. Los demás inquilinos no pueden acceder a las políticas específicas para inquilinos.
Flujo de autorización	Bajo. Todas las consultas se dirigen a un almacén de políticas.	Medio. Debe mantener los mapeos entre cada inquilino y su ID de almacén de pólizas.

Administración de plantillas y esquemas	Bajo. Debe hacer que un esquema funcione para todos los inquilinos.	Alto. Los esquemas y las plantillas pueden ser menos complejos individualmente, pero los cambios requieren más coordinación y complejidad.
Administración de políticas globales	Bajo. Todas las políticas son globales y se pueden actualizar de forma centralizada.	Alto. Debes añadir políticas globales a cada almacén de políticas durante la incorporación. Replica las actualizaciones de las políticas globales entre muchos almacenes de pólizas.
Inquilino abandona el embarque	Alto. Debe identificar y eliminar únicamente las políticas específicas del inquilino.	Bajo. Elimine el almacén de políticas.
Cuotas de recursos de servicio	Altas. Los inquilinos comparten las cuotas de recursos que afectan a los almacenes de políticas, como el tamaño del esquema, el tamaño de las políticas por recurso y las fuentes de identidad por almacén de políticas.	Bajo. Cada inquilino tiene cuotas de recursos específicas.

¿Cómo elegir

Cada aplicación multiusuario es diferente. Compare cuidadosamente los dos enfoques y sus consideraciones antes de tomar una decisión arquitectónica.

Si su aplicación no requiere políticas específicas para cada inquilino y utiliza una única fuente de identidad, es probable que la solución más eficaz sea un almacén de políticas compartido para todos

los inquilinos. Esto se traduce en un flujo de autorización y una gestión de políticas globales más sencillos. La exclusión de un inquilino mediante un almacén de políticas compartido requiere menos esfuerzo, ya que la aplicación no necesita eliminar las políticas específicas del inquilino.

Sin embargo, si su solicitud requiere muchas políticas específicas para cada inquilino o utiliza varias fuentes de identidad, lo más probable es que los almacenes de pólizas por inquilino sean los más eficaces. Puede controlar el acceso a las políticas de inquilinos con IAM políticas que concedan permisos por inquilino a cada almacén de políticas. La exclusión de un inquilino implica eliminar su almacén de pólizas; en un shared-policy-store entorno, debe buscar y eliminar las políticas específicas del inquilino.

Almacenes de políticas de Amazon Verified Permissions

Un almacén de políticas es un contenedor de políticas y plantillas de políticas. En cada almacén de políticas, puede crear un esquema que se utilice para validar las políticas añadidas al almacén de políticas. Además, puede activar la validación de políticas. Si agrega una política a un almacén de políticas con la validación de políticas habilitada, los tipos de entidades, los tipos comunes y las acciones definidas en la política se validan con el esquema y las políticas no válidas se rechazan.

La protección contra la eliminación evita la eliminación accidental de un almacén de políticas. La protección contra la eliminación está habilitada en todos los nuevos almacenes de políticas creados a través de Consola de administración de AWS. Por el contrario, está deshabilitada para todos los almacenes de políticas creados mediante una llamada a la API o al SDK.

Se recomienda crear un almacén de políticas por aplicación o un almacén de políticas por inquilino para las aplicaciones de varios inquilinos. Debe especificar un almacén de políticas al realizar una solicitud de autorización. También puede crear alias de almacenes de políticas para hacer referencia a los almacenes de políticas mediante nombres descriptivos. Para obtener más información, consulte Alias del almacén de políticas de permisos verificados de Amazon.

Le recomendamos que utilice espacios de nombres para las entidades de Cedar en sus almacenes de políticas para evitar la ambigüedad. Un espacio de nombres es un prefijo de cadena para un tipo, separado por un par de signos de dos puntos (`::`) como delimitador. Por ejemplo, `MyApplicationNamespace::exampleType`. Verified Permissions admite hasta 100 espacios de nombres por almacén de políticas. Estos espacios de nombres ayudan a mantener las cosas claras cuando trabajas con varias aplicaciones similares. Por ejemplo, en las aplicaciones con varios inquilinos, si se utiliza un espacio de nombres para añadir el nombre del inquilino a los tipos definidos en el esquema, se diferenciarán de sus homólogos similares utilizados por los demás inquilinos. Al consultar los registros de las solicitudes de autorización, podrá identificar fácilmente al inquilino que procesó la solicitud de autorización. Para obtener información más detallada consulte Espacios de nombres en la Guía de referencia sobre el lenguaje de las políticas de Cedar.

Temas

- Crear almacenes de políticas de Verified Permissions
- Almacenes de políticas vinculados a API
- Eliminar almacenes de políticas

Crear almacenes de políticas de Verified Permissions

Puede crear un almacén de políticas mediante uno de los siguientes métodos:

- Siga una configuración guiada: definirá un tipo de recurso con acciones válidas y un tipo principal antes de crear su primera política.
- Configure con una fuente de identidad API Gateway y una fuente de identidad: defina sus entidades principales con los usuarios que inicien sesión con un proveedor de identidad (IdP) y sus acciones y entidades de recursos desde una API de Amazon API Gateway. Te recomendamos esta opción si quieres que tu aplicación autorice las solicitudes de API con la pertenencia a un grupo de usuarios u otros atributos.
- Comience con un ejemplo de almacén de políticas: elija un ejemplo de almacén de políticas de proyecto predefinido. Le recomendamos esta opción si está aprendiendo sobre Verified Permissions y quiere ver y probar ejemplos de políticas.
- Cree un almacén de políticas vacío: definirá usted mismo el esquema y todas las políticas de acceso. Recomendamos esta opción si ya está familiarizado con la configuración de un almacén de políticas.

Guided setup

Para crear un almacén de políticas mediante Configuración guiada método de configuración

El asistente de configuración guiada le guiará por el proceso de creación de la primera iteración de su almacén de políticas. Creará un esquema para el primer tipo de recurso, describirá las acciones que se aplican a ese tipo de recurso y el tipo de entidad principal para el que va a conceder permisos. A continuación, creará su primera política. Una vez que haya completado este asistente, podrá agregarla a su almacén de políticas, ampliar el esquema para describir otros tipos de recursos y entidades principales y crear políticas y plantillas adicionales.

1. En la consola de permisos verificados, seleccione Crear un nuevo almacén de políticas.
2. En la sección Opciones de inicio, selecciona Configuración guiada.
3. Introduzca una descripción del almacén de políticas. Este texto puede ser el que mejor se adapte a su organización como referencia sencilla a la función del almacén de políticas actual, por ejemplo, la aplicación web de actualizaciones meteorológicas.

4. En la sección Detalles, escriba un espacio de nombres para su esquema. Para obtener más información sobre los espacios de nombres, consulte los espacios de nombres en la documentación de Cedar.
5. Elija Siguiente.
6. En la ventana Tipo de recurso, escriba un nombre para el tipo de recurso. Por ejemplo, `currentTemperature` podría ser un recurso para la aplicación web de actualizaciones meteorológicas.
7. (Opcional) Seleccione Agregar un atributo para añadir los atributos del recurso. Escriba el nombre del atributo y seleccione un tipo de atributo para cada atributo del recurso. Elija si cada atributo es obligatorio. Por ejemplo, `temperatureFormat` podría ser un atributo del `currentTemperature` recurso y estar en grados Fahrenheit o Celsius. Para eliminar un atributo que se ha añadido al tipo de recurso, seleccione Eliminar junto al atributo.
8. En el campo Acciones, escriba las acciones que se van a autorizar para el tipo de recurso especificado. Para agregar acciones adicionales para el tipo de recurso, elija Agregar una acción. Por ejemplo, `viewTemperature` podría ser una acción en la aplicación web de actualizaciones meteorológicas. Para eliminar una acción que se ha añadido al tipo de recurso, seleccione Eliminar junto a la acción.
9. En el campo Nombre del tipo de entidad principal, escriba el nombre del tipo de entidad principal que utilizará las acciones especificadas para el tipo de recurso. De forma predeterminada, el usuario se agrega a este campo, pero se puede reemplazar.
10. Elija Siguiente.
11. En la ventana Tipo de entidad principal, elija la fuente de identidad para su tipo de entidad principal.

    - Elija Personalizado si la aplicación de Verified Permissions proporcionará directamente el ID y los atributos de la entidad principal. Para añadir atributos a la entidad principal, elija Agregar un atributo. Verified Permissions utiliza los valores de atributo especificados al verificar las políticas con el esquema. Para eliminar un atributo que se ha añadido para el tipo principal, seleccione Eliminar junto al atributo.
    - Elija Grupo de usuarios de Cognito si el ID y los atributos del principal se proporcionarán a partir de un ID o token de acceso generado por. Amazon Cognito Seleccione Conectar grupo de usuarios. Seleccione Región de AWSy escriba el ID del grupo de Amazon Cognito usuarios al que desea conectarse. Elija Conectar. Para obtener más información, consulte Autorización con Amazon Verified Permissions en la Guía para desarrolladores de Amazon Cognito.

- Elija un proveedor de OIDC externo si el ID y los atributos del principal se extraerán de un token de and/or acceso de ID generado por un proveedor de OIDC externo y añada los detalles del proveedor y el token.

12. Elija Siguiente.
13. En la sección Detalles de la política, escriba una descripción de la política opcional para su primera política de Cedar.
14. En el campo Ámbito de las entidades principales, elija las entidades principales a las que se les concederán los permisos de la política.

    - Elija Entidad principal específica para aplicar la política a una entidad principal concreta. Elija la entidad principal en el campo Entidad principal a la que se permitirá realizar acciones y escriba un identificador de entidad para la entidad principal. Por ejemplo, `user-id` podría ser un identificador de entidad en la aplicación web Weather updates.

      > **Note**
      >
      > Si lo está utilizando Amazon Cognito, el identificador de la entidad debe tener el formato siguiente`<userpool-id>|<sub>`.

    - Seleccione Todas las entidades principales para aplicar la política a todas las entidades principales de su almacén de políticas.

15. En el campo Ámbito de los recursos, elija los recursos sobre los que las entidades principales especificadas tendrán autorización para actuar.

    - Seleccione Recurso específico para aplicar la política a un recurso específico. Elija el recurso en el campo Recurso al que se debe aplicar esta política y escriba un identificador de entidad para el recurso. Por ejemplo, `temperature-id` podría ser un identificador de entidad en la aplicación web Weather updates.
    - Seleccione Todos los recursos para aplicar la política a todos los recursos de su almacén de políticas.

16. En el campo Ámbito de las acciones, elija las acciones para las que las entidades principales especificadas tendrán autorización para llevar a cabo.

    - Seleccione Conjunto específico de acciones para aplicar la política a acciones concretas. Seleccione las casillas de verificación situadas junto al campo Acciones a las que se debe aplicar esta política.

- Seleccione Todas las acciones para aplicar la política a todas las acciones de su almacén de políticas.

17. Revise la política en la sección Vista previa de la política. Seleccione Crear almacén de políticas.

Set up with API Gateway and an identity source

Para crear un almacén de políticas mediante el Configúrelo con API Gateway y una fuente de identidad método de configuración

La API Gateway opción protege las API con políticas de permisos verificados que están diseñadas para tomar decisiones de autorización a partir de los grupos o roles de los usuarios. Esta opción crea un almacén de políticas para probar la autorización con grupos de fuentes de identidad y una API con un autorizador Lambda.

Los usuarios y sus grupos de un IdP se convierten en sus directores (identificadores) o en su contexto (identificadores de acceso). Los métodos y las rutas de una API Gateway API se convierten en las acciones que autorizan tus políticas. La aplicación se convierte en el recurso. Como resultado de este flujo de trabajo, Verified Permissions crea un almacén de políticas, una función Lambda y un autorizador de API Lambda. Debe asignar el autorizador Lambda a su API después de finalizar este flujo de trabajo.

1. En la consola de permisos verificados, seleccione Crear un nuevo almacén de políticas.
2. En la sección Opciones de inicio, elija Configurar con una fuente de identidad API Gateway y, a continuación, seleccione Siguiente.
3. En el paso Importar recursos y acciones, en API, elige una API que sirva de modelo para los recursos y acciones de tu almacén de políticas.

    a. Elija una etapa de despliegue entre las etapas configuradas en su API y seleccione Importar API. Para obtener más información sobre las etapas de la API, consulte Configuración de una etapa para una API REST en la Guía para desarrolladores de Amazon API Gateway.

    b. Obtenga una vista previa del mapa de recursos y acciones importados.

    c. Para actualizar los recursos o las acciones, modifica las rutas o los métodos de la API en la API Gateway consola y selecciona Importar API para ver las actualizaciones.

    d. Cuando esté satisfecho con sus opciones, seleccione Siguiente.

4. En Origen de identidad, elija un tipo de proveedor de identidad. Puede elegir un grupo de Amazon Cognito usuarios o un tipo de IdP de OpenID Connect (OIDC).
5. Si eligió Amazon Cognito:

   a. Elija un grupo de usuarios en el mismo almacén de políticas Región de AWS y Cuenta de AWS como él.

   b. Elija el tipo de token que desea transferir a la API y que desea enviar para su autorización. Cualquiera de los dos tipos de token contiene grupos de usuarios, que son la base de este modelo de API-linked autorización.

   c. En la validación de clientes de aplicaciones, puede limitar el alcance de un almacén de políticas a un subconjunto de clientes de Amazon Cognito aplicaciones en un grupo de usuarios con varios inquilinos. Para exigir que el usuario se autentique con uno o más clientes de aplicaciones específicos de su grupo de usuarios, seleccione Aceptar solo los tokens con los ID de cliente de aplicación esperados. Para aceptar a cualquier usuario que se autentique en el grupo de usuarios, selecciona No validar los ID de los clientes de la aplicación.

   d. Elija Siguiente.
6. Si has elegido un proveedor de OIDC externo:

   a. En URL del emisor, introduzca la URL de su emisor del OIDC. Este es el punto final del servicio que proporciona, por ejemplo, el servidor de autorización, las claves de firma y otra información sobre su proveedor. `https://auth.example.com` La URL del emisor debe alojar un documento de detección del OIDC en. `/.well-known/openid-configuration`

   b. En Tipo de token, elija el tipo de OIDC JWT que desea que envíe su solicitud de autorización. Para obtener más información, consulte Asignar los Amazon Cognito tokens al esquema y Asignar los tokens OIDC al esquema.

   c. (opcional) En Reclamaciones de token: opcional, selecciona Añadir una notificación de token, introduce un nombre para el token y selecciona un tipo de valor.

   d. En Reclamaciones de token de usuario y grupo, haga lo siguiente:

      i. Introduzca un nombre de reclamo de usuario en el token de la fuente de identidad. Por lo general`sub`, se trata de una afirmación de su ID o token de acceso que contiene el identificador único de la entidad que se va a evaluar. Las identidades del IdP del OIDC conectado se asignarán al tipo de usuario del almacén de políticas.

ii. Introduzca un nombre de reclamación de grupo en el token de la fuente de identidad. Por lo general`groups`, se trata de una afirmación de tu ID o token de acceso que contiene una lista de los grupos de usuarios. El almacén de políticas autorizará las solicitudes en función de la pertenencia al grupo.

e. En la validación de audiencias, elija `Add value` y añada un valor que desee que su almacén de políticas acepte en las solicitudes de autorización.

f. Elija Siguiente.

7. Si así lo desea Amazon Cognito, Verified Permissions consultará los grupos de su grupo de usuarios. En el caso de los proveedores de OIDC, introduzca los nombres de los grupos manualmente. El paso Asignar acciones a los grupos crea políticas para el almacén de políticas que permiten a los miembros del grupo realizar acciones.

   a. Elija o añada los grupos que desee incluir en sus políticas.

   b. Asigna acciones a cada uno de los grupos que has seleccionado.

   c. Elija Siguiente.

8. En Implementar la integración de aplicaciones, elija si desea adjuntar manualmente el autorizador Lambda más adelante o si quiere que Verified Permissions lo haga por usted ahora y revise los pasos que Verified Permissions realizará para crear su almacén de políticas y su autorizador Lambda.

9. Cuando esté listo para crear los nuevos recursos, elija Crear almacén de políticas.

10. Mantén abierto el paso de estado del almacén de políticas en tu navegador para supervisar el progreso de la creación de recursos mediante permisos verificados.

11. Transcurrido algún tiempo, normalmente alrededor de una hora, o cuando el paso Implementar el autorizador Lambda muestre que se ha realizado correctamente, si opta por adjuntar el autorizador manualmente, configure su autorizador.

    Los permisos verificados habrán creado una función de Lambda y un autorizador de Lambda en tu API. Elige Open API para ir a tu API.

    Para obtener información sobre cómo asignar un autorizador de Lambda, consulte Uso de autorizadores de API Gateway Lambda en la Guía para desarrolladores de Amazon API Gateway.

    a. Diríjase a Autorizadores para su API y anote el nombre del autorizador que creó Verified Permissions.

    b. Ve a Recursos y selecciona un método de nivel superior en tu API.

c. Selecciona Editar en la configuración de solicitud de métodos.

d. Configure el autorizador para que sea el nombre del autorizador que anotó anteriormente.

e. Expanda los encabezados de las solicitudes HTTP, introduzca un nombre o y seleccione **`AUTHORIZATION`** Obligatorio.

f. Implemente la etapa de API.

g. Guarde los cambios.

12. Pruebe su autorizador con un token de grupo de usuarios del tipo de token que seleccionó en el paso Elegir la fuente de identidad. Para obtener más información sobre el inicio de sesión del grupo de usuarios y la recuperación de los tokens, consulte el flujo de autenticación del grupo de usuarios en la Guía para desarrolladores de Amazon Cognito.
13. Vuelva a probar la autenticación con un token de grupo de usuarios en el `AUTHORIZATION` encabezado de una solicitud a su API.
14. Examine su nuevo almacén de políticas. Añada y perfeccione las políticas.

Sample policy store

Para crear un almacén de políticas mediante Ejemplo de almacén de políticas método de configuración

1. En la sección Opciones de inicio, elija Sample policy store.
2. En la sección Ejemplo de proyecto, elija el tipo de aplicación de Verified Permissions de muestra que va a utilizar.

   - PhotoFlashes un ejemplo de aplicación web orientada al cliente que permite a los usuarios compartir fotos y álbumes individuales con amigos. Los usuarios pueden establecer permisos detallados sobre quién puede ver, comentar y volver a compartir sus fotos. Los propietarios de las cuentas también pueden crear grupos de amigos y organizar las fotos en álbumes.
   - DigitalPetStorees un ejemplo de aplicación en el que cualquiera puede registrarse y convertirse en cliente. Los clientes pueden añadir mascotas para vender, buscar mascotas y realizar pedidos. Los clientes que han añadido una mascota se registran como propietarios de la mascota. Los dueños de mascotas pueden actualizar los detalles de la mascota, subir una imagen de ella o eliminar la lista de mascotas. Los clientes que han realizado un pedido quedan registrados como propietarios del pedido. Los propietarios

de los pedidos pueden obtener los detalles del pedido o cancelarlo. Los gerentes de las tiendas de mascotas tienen acceso de administrador.

> **Note**
>
> El almacén de políticas de DigitalPetStoremuestra no incluye plantillas de políticas. Los almacenes TinyTododе políticas PhotoFlashy los de muestra incluyen plantillas de políticas.

- TinyTodoes una aplicación de ejemplo que permite a los usuarios crear tareas y listas de tareas. Los propietarios de las listas pueden administrar y compartir sus listas y especificar quién puede verlas o editarlas.

3. Se generará automáticamente un espacio de nombres para el esquema del almacén de políticas de muestra en función del proyecto de ejemplo que haya elegido.
4. Seleccione Crear almacén de políticas.

   El almacén de políticas se crea con políticas y un esquema para el almacén de políticas de muestra que elija. Para obtener más información sobre las políticas vinculadas a plantillas que puede crear para los almacenes de políticas de muestra, consulte Ejemplos de políticas vinculadas a plantillas de permisos verificados de Amazon.

Empty policy store

Para crear un almacén de políticas mediante Almacén de políticas vacío método de configuración

1. En la sección Opciones de inicio, elija Vacía el almacén de políticas.
2. Seleccione Crear almacén de políticas.

Se crea un almacén de políticas vacío sin un esquema, lo que significa que las políticas no se validan. Para obtener más información acerca de cómo actualizar el esquema del almacén de políticas, consulte Esquema del almacén de políticas de Amazon Verified Permissions..

Para obtener más información sobre cómo crear políticas para su almacén de políticas, consulte Creación de políticas estáticas de Amazon Verified Permissions y Creación de políticas vinculadas a plantillas de permisos verificados de Amazon.

AWS CLI

Para crear un almacén de políticas vacío mediante el AWS CLI.

Puede crear un almacén de políticas mediante la operación `create-policy-store`.

> **Note**
>
> Un almacén de políticas que se crea mediante el AWS CLI está vacío.
>
> - Para añadir un esquema, consulte Esquema del almacén de políticas de Amazon Verified Permissions..
> - Para añadir políticas, consulte Creación de políticas estáticas de Amazon Verified Permissions.
> - Para añadir plantillas de políticas, consulte Creación de plantillas de políticas de permisos verificados de Amazon.

```
$ aws verifiedpermissions create-policy-store \
    --validation-settings "mode=STRICT"
{
    "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111",
    "createdDate": "2023-05-16T17:41:29.103459+00:00",
    "lastUpdatedDate": "2023-05-16T17:41:29.103459+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

AWS SDKs

Puede crear un almacén de políticas mediante la API `CreatePolicyStore`. Para obtener más información, consulta la Guía CreatePolicyStorede referencia de la API de permisos verificados de Amazon.

Implementación de los permisos verificados de Amazon en Rust con AWS SDK

En este tema se proporciona un ejemplo práctico de cómo implementar los permisos verificados de Amazon en Rust con el AWS SDK. En este ejemplo se muestra cómo desarrollar un modelo de

autorización que pueda comprobar si un usuario puede ver una foto. El código de ejemplo utiliza la caja aws-sdk-verifiedpermissions de the AWS SDK para Rust, que ofrece un conjunto sólido de herramientas con las que interactuar. Servicios de AWS

Requisitos previos

Antes de empezar, asegúrese de que tiene la AWS CLI configurada en su sistema y de que está familiarizado con Rust.

- Para obtener instrucciones sobre la instalación AWS CLI, consulte la guía de instalación de la AWS CLI.
- Para obtener instrucciones sobre la configuración AWS CLI, consulte Configuración de los ajustes para AWS CLI y los ajustes del archivo de configuración y credenciales en. AWS CLI
- Para obtener más información sobre Rust, consulte rust-lang.org y la Guía para desarrolladores del AWS SDK para Rust.

Con su entorno preparado, exploremos cómo implementar los permisos verificados en Rust.

Pruebe el código de muestra

El código de ejemplo hace lo siguiente:

- Configura el cliente del SDK con el que debe comunicarse AWS
- Crea un almacén de políticas
- Define la estructura del almacén de políticas añadiendo un esquema
- Añade una política para comprobar las solicitudes de autorización
- Envía una solicitud de autorización de prueba para comprobar que todo está configurado correctamente

Para probar el código de muestra

1. Crea un proyecto de Rust.
2. Sustituya cualquier código existente por el siguiente código: `main.rs`

```
use std::time::Duration;
use std::thread::sleep;
use aws_config::BehaviorVersion;
use aws_sdk_verifiedpermissions::Client;
```

```
use aws_sdk_verifiedpermissions::{
    operation::{
        create_policy::CreatePolicyOutput,
        create_policy_store::CreatePolicyStoreOutput,
        is_authorized::IsAuthorizedOutput,
        put_schema::PutSchemaOutput,
    },
    types::{
        ActionIdentifier, EntityIdentifier, PolicyDefinition, SchemaDefinition,
 StaticPolicyDefinition, ValidationSettings
    },
};

//Function that creates a policy store in the client that's passed
async fn create_policy_store(client: &Client, valid_settings: &ValidationSettings)-
> CreatePolicyStoreOutput {
    let policy_store =
 client.create_policy_store().validation_settings(valid_settings.clone()).send().await;
    return policy_store.unwrap();
}

//Function that adds a schema to the policy store in the client
async fn put_schema(client: &Client, ps_id: &str, schema: &str) -> PutSchemaOutput
 {
    let schema =
 client.put_schema().definition(SchemaDefinition::CedarJson(schema.to_string())).policy_sto
    return schema.unwrap();
}

//Function that creates a policy in the policy store in the client
async fn create_policy(client: &Client, ps_id: &str,
 policy_definition:&PolicyDefinition) -> CreatePolicyOutput {
    let create_policy =
 client.create_policy().definition(policy_definition.clone()).policy_store_id(ps_id).send()
    return create_policy.unwrap();
}

//Function that tests the authorization request to the policy store in the client
async fn authorize(client: &Client, ps_id: &str, principal: &EntityIdentifier,
 action: &ActionIdentifier, resource: &EntityIdentifier) -> IsAuthorizedOutput {
    let is_auth =
 client.is_authorized().principal(principal.to_owned()).action(action.to_owned()).resource(
    return is_auth.unwrap();
}
```

```
#[::tokio::main]
async fn main() -> Result<(), aws_sdk_verifiedpermissions::Error> {

//Set up SDK client
    let config = aws_config::load_defaults(BehaviorVersion::latest()).await;
    let client = aws_sdk_verifiedpermissions::Client::new(&config);

//Create a policy store
    let valid_settings = ValidationSettings::builder()
    .mode({aws_sdk_verifiedpermissions::types::ValidationMode::Strict
    })
    .build()
    .unwrap();
    let policy_store = create_policy_store(&client, &valid_settings).await;
    println!(
    "Created Policy store with ID: {:?}",
    policy_store.policy_store_id
    );

//Add schema to policy store
    let schema= r#"{
        "PhotoFlash": {
            "actions": {
                "ViewPhoto": {
                    "appliesTo": {
                        "context": {
                            "type": "Record",
                            "attributes": {}
                        },
                        "principalTypes": [
                            "User"
                        ],
                        "resourceTypes": [
                            "Photo"
                        ]
                    },
                    "memberOf": []
                }
            },
            "entityTypes": {
                "Photo": {
                    "memberOfTypes": [],
                    "shape": {
```

```
                        "type": "Record",
                        "attributes": {
                            "IsPrivate": {
                                "type": "Boolean"
                            }
                        }
                    }
                },
                "User": {
                    "memberOfTypes": [],
                    "shape": {
                        "attributes": {},
                        "type": "Record"
                    }
                }
            }
        }
    }"#;
    let put_schema = put_schema(&client, &policy_store.policy_store_id,
 schema).await;
    println!(
        "Created Schema with Namespace: {:?}",
        put_schema.namespaces
    );

//Create policy
    let policy_text = r#"
        permit (
            principal in PhotoFlash::User::"alice",
            action == PhotoFlash::Action::"ViewPhoto",
            resource == PhotoFlash::Photo::"VacationPhoto94.jpg"
        );
        "#;
    let policy_definition =
 PolicyDefinition::Static(StaticPolicyDefinition::builder().statement(policy_text).build().
    let policy = create_policy(&client, &policy_store.policy_store_id,
 &policy_definition).await;
    println!(
        "Created Policy with ID: {:?}",
        policy.policy_id
    );

//Break to make sure the resources are created before testing authorization
    sleep(Duration::new(2, 0));
```

```
//Test authorization
    let principal=
 EntityIdentifier::builder().entity_id("alice").entity_type("PhotoFlash::User").build().unw
    let action =
 ActionIdentifier::builder().action_type("PhotoFlash::Action").action_id("ViewPhoto").build
    let resource =
 EntityIdentifier::builder().entity_id("VacationPhoto94.jpg").entity_type("PhotoFlash::Phot
    let auth = authorize(&client, &policy_store.policy_store_id, &principal,
 &action, &resource).await;
    println!(
        "Decision: {:?}",
        auth.decision
        );
        println!(
        "Policy ID: {:?}",
        auth.determining_policies
        );
     Ok(())
}
```

3. Ejecuta el código ingresándolo `cargo run` en la terminal.

Si el código se ejecuta correctamente, aparecerá el terminal `Decision: Allow` seguido del identificador de la política determinante. Esto significa que ha creado correctamente un almacén de políticas y lo ha probado con el AWS SDK de Rust.

Eliminar recursos

Cuando hayas terminado de explorar tu almacén de políticas, elimínalo.

Para eliminar un almacén de políticas

Puedes eliminar un almacén de políticas mediante la `delete-policy-store` operación y *`PSEXAMPLEabcdefg111111`* sustituirlo por el identificador del almacén de políticas que deseas eliminar.

```
$ aws verifiedpermissions delete-policy-store \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Si se ejecuta correctamente, este comando no genera ninguna salida.

Almacenes de políticas vinculados a API

Un caso de uso común es utilizar los permisos verificados de Amazon para autorizar el acceso de los usuarios a los APIs sitios alojados en Amazon API Gateway. Con un asistente en la AWS consola, puede crear políticas de acceso basadas en roles para los usuarios administrados en Amazon Cognito o en cualquier proveedor de identidad (IdP) de OIDC e AWS Lambda implementar un autorizador que llame a Verified Permissions para evaluar estas políticas.

Para completar el asistente, elija Configurar con un proveedor de identidad al crear un nuevo almacén de políticas API Gateway y siga los pasos.

Se crea un almacén de políticas vinculado a una API que aprovisiona el modelo de autorización y los recursos para las solicitudes de autorización. El almacén de políticas tiene una fuente de identidad y un autorizador Lambda que se conecta API Gateway a Verified Permissions. Una vez creado el almacén de políticas, puede autorizar las solicitudes de API en función de la pertenencia a grupos de usuarios. Por ejemplo, los permisos verificados solo pueden conceder acceso a los usuarios que son miembros del `Directors` grupo.

A medida que su aplicación crezca, podrá implementar una autorización detallada con atributos de usuario y ámbitos OAuth 2.0 utilizando el lenguaje de políticas de Cedar. Por ejemplo, los permisos verificados solo pueden conceder acceso a los usuarios que tienen un `email` atributo en el dominio. `mycompany.co.uk`

Una vez que haya configurado el modelo de autorización para su API, su responsabilidad restante es autenticar a los usuarios y generar las solicitudes de API en su aplicación, así como mantener su almacén de políticas.

Para ver una demostración, consulta Amazon Verified Permissions: Quick Start Overview y Demo en el Amazon Web Services YouTube canal.

Temas

- Cómo autoriza Verified Permissions las solicitudes de API
- Consideraciones sobre los almacenes de políticas vinculados a la API
- Añadir un control de acceso basado en atributos (ABAC)
- Pasar a la producción con AWS CloudFormation
- Solución de problemas de almacenes de políticas vinculados a API

> **Important**
>
> Los almacenes de políticas que se crean con la opción Configurar con API Gateway y una fuente de identidad en la consola de permisos verificados no están pensados para su despliegue inmediato en la producción. Con el almacén de políticas inicial, finalice el modelo de autorización y exporte los recursos del almacén de políticas al CloudFormation. Implemente los permisos verificados en la producción mediante programación con. Kit de desarrollo de la nube de AWS (AWS CDK) Para obtener más información, consulte Pasar a la producción con AWS CloudFormation.

En un almacén de políticas vinculado a una API y a una fuente de identidad, tu aplicación presenta un token de grupo de usuarios en un encabezado de autorización cuando realiza una solicitud a la API. La fuente de identidad de tu almacén de políticas proporciona la validación de los tokens para los permisos verificados. El token forma las solicitudes `principal` de autorización de entrada con la IsAuthorizedWithTokenAPI. Los permisos verificados crean políticas en torno a la pertenencia a grupos de sus usuarios, tal como se presenta en la declaración de un grupo en términos de identidad (ID) y de acceso, por `cognito:groups` ejemplo, en el caso de los grupos de usuarios. Su API procesa el token de su aplicación en un autorizador Lambda y lo envía a Verified Permissions para que tome una decisión de autorización. Cuando su API recibe la decisión de autorización del autorizador de Lambda, transfiere la solicitud a su fuente de datos o la deniega.

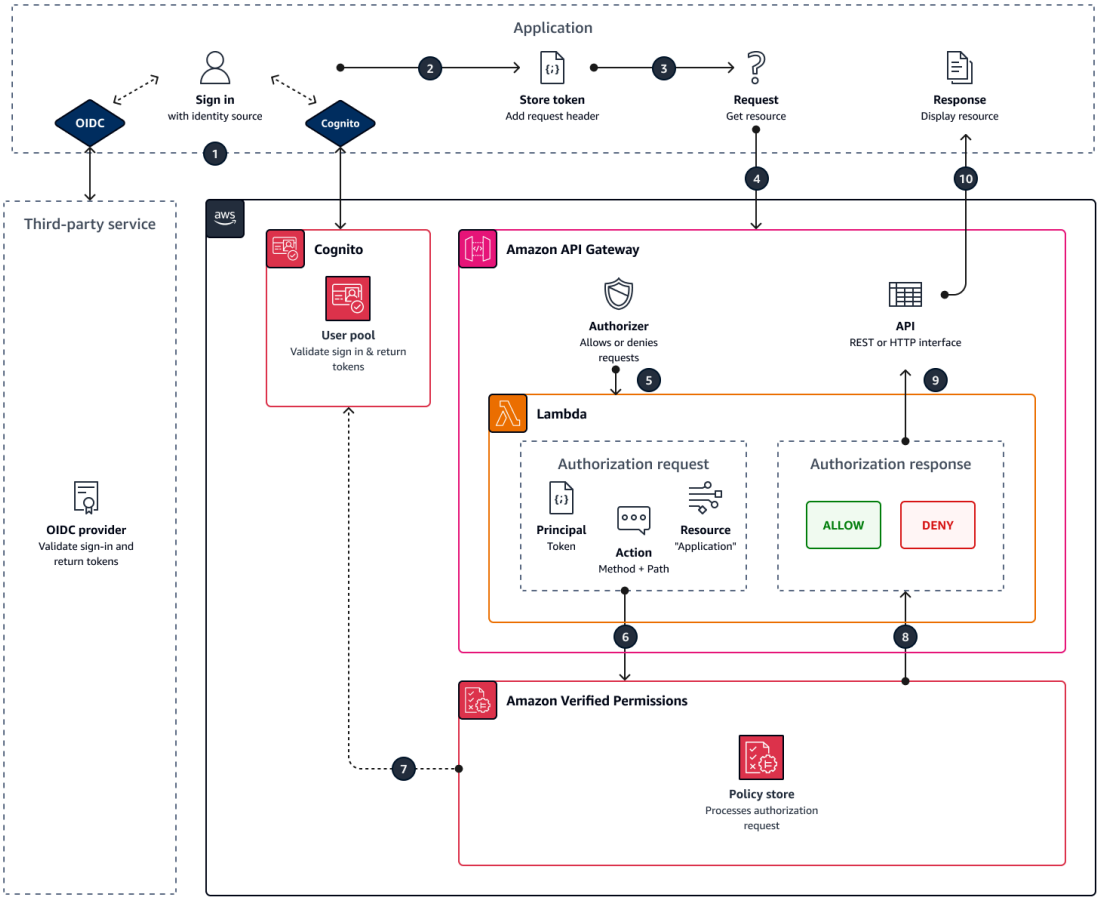
Componentes de la identidad, la fuente y la API Gateway autorización con permisos verificados

- Un grupo Amazon Cognitode usuarios o un IdP de OIDC que autentica y agrupa a los usuarios. Los tokens de los usuarios completan la pertenencia al grupo y el principal o el contexto que Verified Permissions evalúa en su almacén de políticas.
- Una API API GatewayREST. Los permisos verificados definen las acciones a partir de las rutas y los métodos de la API, por ejemplo`MyAPI::Action::get /photo`.
- Una función de Lambda y un autorizador de Lambda para su API. La función Lambda toma los tokens portadores del grupo de usuarios, solicita la autorización de Verified Permissions y devuelve una decisión a. API Gateway El flujo de trabajo Configurar con API Gateway una fuente de identidad crea automáticamente este autorizador Lambda por usted.
- Un almacén de políticas de permisos verificados. La fuente de identidad del almacén de políticas es su grupo Amazon Cognito de usuarios o grupo de proveedores de OIDC. El esquema del almacén de políticas refleja la configuración de la API y las políticas vinculan a los grupos de usuarios con las acciones de la API permitidas.

- Una aplicación que autentica a los usuarios con tu IDP y añade tokens a las solicitudes de la API.

Cómo autoriza Verified Permissions las solicitudes de API

Al crear un nuevo almacén de políticas y seleccionar la opción Configurar con API Gateway una fuente de identidad, Verified Permissions crea el esquema y las políticas del almacén de políticas. El esquema y las políticas reflejan las acciones de la API y los grupos de usuarios a los que desea autorizar para que las realicen. Verified Permissions también crea la función Lambda y el autorizador.



1. El usuario inicia sesión con la aplicación a través de Amazon Cognito u otro IdP del OIDC. El IdP emite identificadores de acceso y de identificación con la información del usuario.
2. Su aplicación almacena el JWTs. Para obtener más información, consulte Uso de tokens con grupos de usuarios en la Guía para Amazon Cognito desarrolladores.
3. El usuario solicita datos que la aplicación debe recuperar de una API externa.
4. Su aplicación solicita datos de una API REST en API Gateway. Añade un ID o un token de acceso como encabezado de solicitud.

5. Si tu API tiene una memoria caché para la decisión de autorización, devuelve la respuesta anterior. Si el almacenamiento en caché está deshabilitado o la API no tiene caché actual, API Gateway pasa los parámetros de la solicitud a un autorizador Lambda basado en tokens.
6. La función Lambda envía una solicitud de autorización a un almacén de políticas de permisos verificados con la IsAuthorizedWithTokenAPI. La función Lambda transmite los elementos de una decisión de autorización:
   a. El token del usuario como principal.
   b. El método de la API combinado con la ruta de la API`GetPhoto`, por ejemplo, como acción.
   c. El término `Application` como recurso.
7. Los permisos verificados validan el token. Para obtener más información sobre cómo se validan los Amazon Cognito tokens, consulte Autorización con permisos verificados de Amazon en la Guía para Amazon Cognito desarrolladores.
8. Verified Permissions evalúa la solicitud de autorización comparándola con las políticas de tu almacén de políticas y devuelve una decisión de autorización.
9. El autorizador Lambda devuelve una respuesta `Allow` or `Deny` a. API Gateway
10. La API devuelve datos o una `ACCESS_DENIED` respuesta a su aplicación. La aplicación procesa y muestra los resultados de la solicitud de API.

Consideraciones sobre los almacenes de políticas vinculados a la API

Al crear un almacén de políticas vinculado a una API en la consola de permisos verificados, se crea una prueba para una posible implementación en producción. Antes de pasar a la fase de producción, establece una configuración fija para la API y el grupo de usuarios. Tenga en cuenta los siguientes factores:

API Gateway almacena en caché las respuestas

> En los almacenes de políticas vinculados a API, Verified Permissions crea un autorizador Lambda con un TTL de almacenamiento en caché de autorización de 120 segundos. Puede ajustar este valor o desactivar el almacenamiento en caché en su autorizador. En un autorizador con el almacenamiento en caché activado, el autorizador devuelve la misma respuesta cada vez hasta que caduque el TTL. Esto puede prolongar la vida útil efectiva de los tokens del grupo de usuarios hasta un tiempo igual al TTL de almacenamiento en caché de la etapa solicitada.

Amazon Cognito los grupos se pueden reutilizar

Amazon Verified Permissions determina la pertenencia a un grupo para los usuarios del grupo de usuarios a partir de la `cognito:groups` declaración que figura en el identificador de usuario o en el token de acceso. El valor de esta afirmación es un conjunto de nombres descriptivos de los grupos de usuarios a los que pertenece el usuario. No puede asociar grupos de grupos de usuarios con un identificador único.

Los grupos de usuarios que se eliminan y se vuelven a crear con el mismo nombre se presentan en el almacén de políticas como el mismo grupo. Al eliminar un grupo de un grupo de usuarios, elimine todas las referencias al grupo del almacén de políticas.

El espacio de nombres y el esquema derivados de la API son point-in-time

Verified Permissions captura tu API en un momento dado: solo consulta tu API cuando creas tu almacén de políticas. Cuando el esquema o el nombre de la API cambien, debe actualizar el almacén de políticas y el autorizador de Lambda, o bien crear un nuevo almacén de políticas vinculado a la API. Verified Permissions deriva el espacio de nombres del almacén de políticas del nombre de tu API.

La función Lambda no tiene configuración de VPC

La función Lambda que Verified Permissions crea para su autorizador de API se lanza en la VPC predeterminada. De forma predeterminada. APIs los que tienen acceso a la red restringido a privado no VPCs pueden comunicarse con la función Lambda que autoriza las solicitudes de acceso con permisos verificados.

Verified Permissions implementa los recursos del autorizador en CloudFormation

Para crear un almacén de políticas vinculado a una API, debe iniciar sesión con un AWS director con muchos privilegios en la consola de permisos verificados. Este usuario despliega una CloudFormation pila que crea recursos en varios. Servicios de AWS Este director debe tener el permiso para agregar y modificar recursos en Verified Permissions IAM, Lambda y. API Gateway Como práctica recomendada, no comparta estas credenciales con otros administradores de su organización.

Consulte Pasar a la producción con AWS CloudFormation para obtener una descripción general de los recursos que crea Verified Permissions.

Añadir un control de acceso basado en atributos (ABAC)

Una sesión de autenticación típica con un IdP devuelve los identificadores de identificación y acceso. Puedes pasar cualquiera de estos tipos de token como token portador en las solicitudes de aplicación a tu API. En función de lo que elijas al crear tu almacén de políticas, Verified Permissions espera uno de los dos tipos de token. Ambos tipos contienen información sobre la pertenencia al grupo del usuario. Para obtener más información sobre los tipos de token Amazon Cognito, consulte Uso de tokens con grupos de usuarios en la Guía para Amazon Cognito desarrolladores.

Tras crear un almacén de políticas, puede añadir y ampliar políticas. Por ejemplo, puede agregar nuevos grupos a sus políticas a medida que los agrega a su grupo de usuarios. Como su almacén de políticas ya conoce la forma en que su grupo de usuarios presenta los grupos en forma de tokens, puede permitir un conjunto de acciones para cualquier grupo nuevo con una política nueva.

Es posible que también desee ampliar el modelo de evaluación de políticas basado en grupos para convertirlo en un modelo más preciso basado en las propiedades de los usuarios. Los tokens del grupo de usuarios contienen información de usuario adicional que puede contribuir a las decisiones de autorización.

Tokens de ID

Los tokens de identificación representan los atributos de un usuario y tienen un alto nivel de control de acceso detallado. Para evaluar las direcciones de correo electrónico, los números de teléfono o los atributos personalizados, como el departamento y el gerente, evalúa el token de identificación.

Tokens de acceso

Los tokens de acceso representan los permisos de un usuario con un alcance OAuth 2.0. Para añadir una capa de autorización o configurar solicitudes de recursos adicionales, evalúa el token de acceso. Por ejemplo, puedes validar que un usuario esté en los grupos adecuados y tenga un ámbito como el `PetStore.read` que generalmente autoriza el acceso a la API. Los grupos de usuarios pueden añadir ámbitos personalizados a los tokens con servidores de recursos y con la personalización de los mismos durante el tiempo de ejecución.

Consulte Asignar Amazon Cognito los tokens al esquema y Asignar los tokens de OIDC al esquema para ver, por ejemplo, las políticas que procesan las reclamaciones en los tokens de identificación y acceso.

Pasar a la producción con AWS CloudFormation

Los almacenes de políticas vinculados a las API son una forma de crear rápidamente un modelo de autorización para una API. API Gateway Están diseñados para servir como entorno de pruebas para el componente de autorización de la aplicación. Después de crear su almacén de políticas de prueba, dedique tiempo a refinar las políticas, el esquema y el autorizador Lambda.

Puede ajustar la arquitectura de la API y requerir ajustes equivalentes en el esquema y las políticas del almacén de políticas. Los almacenes de políticas vinculados a las API no actualizan automáticamente su esquema desde la arquitectura de la API: Verified Permissions solo consulta la API en el momento de crear un almacén de políticas. Si tu API cambia lo suficiente, es posible que tengas que repetir el proceso con un nuevo almacén de políticas.

Cuando su modelo de aplicación y autorización estén listos para su implementación en producción, integre el almacén de políticas vinculado a la API que desarrolló con sus procesos de automatización. Como práctica recomendada, le recomendamos que exporte el esquema y las políticas del almacén de políticas a una AWS CloudFormation plantilla que pueda implementar en otras Cuentas de AWS . Regiones de AWS

Los resultados del proceso del almacén de políticas vinculado a la API son un almacén de políticas inicial y un autorizador Lambda. El autorizador Lambda tiene varios recursos dependientes. Verified Permissions implementa estos recursos en una pila generada automáticamente. CloudFormation Para implementarlo en producción, debe recopilar el almacén de políticas y los recursos del autorizador Lambda en una plantilla. Un almacén de políticas vinculado a una API se compone de los siguientes recursos:

1. AWS::VerifiedPermissions::PolicyStore: Copie el esquema en el `SchemaDefinition` objeto. Escapa de " los personajes como\".
2. AWS::VerifiedPermissions::IdentitySource: Copie los valores de la salida de su almacén GetIdentitySource de políticas de prueba y modifíquelos según sea necesario.
3. Una o más de las AWS::VerifiedPermissions::Policysiguientes opciones: Copie la declaración de política en el `Definition` objeto. Escapa de " los personajes como\".
4. AWS: :Lambda: :Función, AWS::: Función,IAM:::Política,AWS::IAM: :Autorizador,AWS ApiGateway AWS::Lambda::Permission

La siguiente plantilla es un ejemplo de almacén de políticas. Puede añadir los recursos del autorizador Lambda de su pila existente a esta plantilla.

```
{
    "AWSTemplateFormatVersion": "2010-09-09",
    "Resources": {
        "MyExamplePolicyStore": {
            "Type": "AWS::VerifiedPermissions::PolicyStore",
            "Properties": {
                "ValidationSettings": {
                    "Mode": "STRICT"
                },
                "Description": "ApiGateway: PetStore/test",
                "Schema": {
                    "CedarJson": "{\"PetStore\":{\"actions\":{\"get /pets\":
{\"appliesTo\":{\"principalTypes\":[\"User\"],\"resourceTypes\":[\"Application\"],
\"context\":{\"type\":\"Record\",\"attributes\":{}}}},\"get /\":{\"appliesTo\":
{\"principalTypes\":[\"User\"],\"resourceTypes\":[\"Application\"],\"context\":{\"type
\":\"Record\",\"attributes\":{}}}},\"get /pets/{petId}\":{\"appliesTo\":{\"context
\":{\"type\":\"Record\",\"attributes\":{}},\"resourceTypes\":[\"Application\"],
\"principalTypes\":[\"User\"]}},\"post /pets\":{\"appliesTo\":{\"principalTypes\":
[\"User\"],\"resourceTypes\":[\"Application\"],\"context\":{\"type\":\"Record\",
\"attributes\":{}}}}},\"entityTypes\":{\"Application\":{\"shape\":{\"type\":\"Record\",
\"attributes\":{}}},\"User\":{\"memberOfTypes\":[\"UserGroup\"],\"shape\":{\"attributes
\":{},\"type\":\"Record\"}},\"UserGroup\":{\"shape\":{\"type\":\"Record\",\"attributes
\":{}}}}}}"
                }
            }
        },
        "MyExamplePolicy": {
            "Type": "AWS::VerifiedPermissions::Policy",
            "Properties": {
                "Definition": {
                    "Static": {
                        "Description": "Policy defining permissions for testgroup
 cognito group",
                        "Statement": "permit(\nprincipal in PetStore::UserGroup::
\"us-east-1_EXAMPLE|testgroup\",\naction in [\n  PetStore::Action::\"get /\",
\n  PetStore::Action::\"post /pets\",\n  PetStore::Action::\"get /pets\",\n
 PetStore::Action::\"get /pets/{petId}\"\n],\nresource);"
                    }
                },
                "PolicyStoreId": {
                    "Ref": "MyExamplePolicyStore"
                }
            },
```

```
            "DependsOn": [
                "MyExamplePolicyStore"
            ]
        },
        "MyExampleIdentitySource": {
            "Type": "AWS::VerifiedPermissions::IdentitySource",
            "Properties": {
                "Configuration": {
                    "CognitoUserPoolConfiguration": {
                        "ClientIds": [
                            "1example23456789"
                        ],
                        "GroupConfiguration": {
                            "GroupEntityType": "PetStore::UserGroup"
                        },
                        "UserPoolArn": "arn:aws:cognito-idp:us-
east-1:123456789012:userpool/us-east-1_EXAMPLE"
                    }
                },
                "PolicyStoreId": {
                    "Ref": "MyExamplePolicyStore"
                },
                "PrincipalEntityType": "PetStore::User"
            },
            "DependsOn": [
                "MyExamplePolicyStore"
            ]
        }
    }
}
```

Solución de problemas de almacenes de políticas vinculados a API

Usa la información aquí para ayudarte a diagnosticar y solucionar problemas comunes al crear almacenes de políticas vinculados a la API de Amazon Verified Permissions.

Temas

- He actualizado mi política, pero la decisión de autorización no ha cambiado
- He adjuntado el autorizador Lambda a mi API, pero no genera solicitudes de autorización
- He recibido una decisión de autorización inesperada y quiero revisar la lógica de autorización
- Quiero buscar los registros de mi autorizador Lambda

- Mi autorizador Lambda no existe
- Mi API está en una VPC privada y no puedo invocar el autorizador
- Quiero procesar atributos de usuario adicionales en mi modelo de autorización
- Quiero añadir nuevas acciones, atributos de contexto de acción o atributos de recursos

He actualizado mi política, pero la decisión de autorización no ha cambiado

De forma predeterminada, Verified Permissions configura el autorizador Lambda para almacenar en caché las decisiones de autorización durante 120 segundos. Vuelva a intentarlo transcurridos dos minutos o desactive la memoria caché de su autorizador. Para obtener más información, consulte Habilitar el almacenamiento en caché de las API para mejorar la capacidad de respuesta en la Guía para desarrolladores de Amazon API Gateway.

He adjuntado el autorizador Lambda a mi API, pero no genera solicitudes de autorización

Para empezar a procesar las solicitudes, debe implementar la etapa de API a la que adjuntó su autorizador. Para obtener más información, consulte Implementación de una API REST en la Guía para desarrolladores de Amazon API Gateway.

He recibido una decisión de autorización inesperada y quiero revisar la lógica de autorización

El proceso del almacén de políticas vinculado a la API crea una función Lambda para el autorizador. Verified Permissions incorpora automáticamente la lógica de sus decisiones de autorización a la función de autorización. Después de crear el almacén de políticas, puede volver atrás para revisar y actualizar la lógica de la función.

Para localizar la función Lambda desde la AWS CloudFormation consola, pulse el botón Comprobar despliegue en la página de descripción general del nuevo almacén de políticas.

También puede localizar la función en la AWS Lambda consola. Navegue hasta la consola en el almacén Región de AWS de políticas y busque el nombre de una función con el prefijo de`AVPAuthorizerLambda`. Si ha creado más de un almacén de políticas vinculado a una API, utilice la hora de la última modificación de sus funciones para correlacionarlas con la creación del almacén de políticas.

Quiero buscar los registros de mi autorizador Lambda

Las funciones Lambda recopilan métricas y registran sus resultados de invocación en Amazon. CloudWatch Para revisar los registros, localice la función en la consola de Lambda y seleccione la pestaña Supervisar. Seleccione Ver CloudWatch registros y revise las entradas del grupo de registros.

Para obtener más información sobre los registros de funciones de Lambda, consulte Uso de Amazon CloudWatch Logs con AWS Lambda en la Guía para AWS Lambda desarrolladores.

Mi autorizador Lambda no existe

Tras completar la configuración de un almacén de políticas vinculado a la API, debe adjuntar el autorizador Lambda a la API. Si no encuentra su autorizador en la API Gateway consola, es posible que los recursos adicionales de su almacén de políticas hayan fallado o que aún no se hayan implementado. Los almacenes de políticas vinculados a las API implementan estos recursos en una pila. CloudFormation

Verified Permissions muestra un enlace con la etiqueta Comprobar el despliegue al final del proceso de creación. Si ya has salido de esta pantalla, ve a la CloudFormation consola y busca en las pilas recientes un nombre que lleve el prefijo. `AVPAuthorizer-<policy store ID>` CloudFormation proporciona información valiosa sobre la solución de problemas en el resultado de una implementación de stack.

Para obtener ayuda con la solución de problemas de CloudFormation pilas, consulte Solución de problemas CloudFormation en la Guía del AWS CloudFormation usuario.

Mi API está en una VPC privada y no puedo invocar el autorizador

Los permisos verificados no admiten el acceso a los autorizadores de Lambda a través de puntos de enlace de VPC. Debe abrir una ruta de red entre su API y la función Lambda que actúa como su autorizador.

Quiero procesar atributos de usuario adicionales en mi modelo de autorización

El proceso de almacenamiento de políticas vinculado a la API deriva las políticas de permisos verificados de las declaraciones del grupo en los tokens de los usuarios. Para actualizar su modelo de autorización y tener en cuenta otros atributos de usuario, integre esos atributos en sus políticas.

Puede asignar muchas reclamaciones de los tokens de ID y acceso de los grupos de usuarios de Amazon Cognito a las declaraciones de la política de permisos verificados. Por ejemplo, la mayoría

de los usuarios tienen una `email` reclamación en su token de identificación. Para obtener más información sobre cómo añadir notificaciones de tu fuente de identidad a las políticas, consulta Asignar los Amazon Cognito tokens al esquema y Asignar los tokens OIDC al esquema.

Quiero añadir nuevas acciones, atributos de contexto de acción o atributos de recursos

Un almacén de políticas vinculado a una API y el autorizador Lambda que crea son un recurso. point-in-time Reflejan el estado de la API en el momento de su creación. El esquema del almacén de políticas no asigna ningún atributo de contexto a las acciones, ni ningún atributo o elemento principal al `Application` recurso predeterminado.

Cuando agregas acciones (rutas y métodos) a tu API, debes actualizar tu almacén de políticas para estar al tanto de las nuevas acciones. También debe actualizar su autorizador Lambda para procesar las solicitudes de autorización para las nuevas acciones. Puede empezar de nuevo con un almacén de políticas nuevo o actualizar el almacén de políticas existente.

Para actualizar tu almacén de políticas existente, localiza tu función. Examine la lógica de la función generada automáticamente y actualícela para procesar las nuevas acciones, atributos o contextos. A continuación, edite el esquema para incluir las nuevas acciones y atributos.

Eliminar almacenes de políticas

Puedes eliminar los almacenes de políticas de permisos verificados de Amazon mediante el Consola de administración de AWS o el AWS CLI. Al eliminar un almacén de políticas, se eliminan permanentemente el esquema y todas las políticas y plantillas de políticas del almacén de políticas. También se eliminarán todos los alias del almacén de políticas asociado al almacén de políticas eliminado.

La protección contra la eliminación evita la eliminación accidental de un almacén de políticas. La protección contra la eliminación está habilitada en todos los nuevos almacenes de políticas creados a través de Consola de administración de AWS. Por el contrario, está deshabilitada para todos los almacenes de políticas creados mediante una llamada a la API o al SDK.

Es posible que desee eliminar los almacenes de políticas por los siguientes motivos:

- Ha alcanzado la cuota de almacenes de pólizas disponibles en una región determinada. Para obtener más información, consulte Cuotas de recursos.

- Ya no apoyas a un inquilino en una solicitud con varios inquilinos y, por lo tanto, ya no necesitas ese almacén de pólizas.

Consola de administración de AWS

Para eliminar un almacén de políticas

1. Abre la consola de permisos verificados. Elige tu almacén de políticas.
2. En el panel de navegación izquierdo, elija Configuración.
3. Seleccione Eliminar este almacén de políticas.
4. Escriba `delete` en la casilla de texto y seleccione Eliminar.

> **Note**
>
> Si la protección contra la eliminación está habilitada, tendrás que deshabilitarla antes de poder elegir Eliminar. Para desactivarla, selecciona Desactivar la protección contra eliminaciones.

AWS CLI

Para eliminar un almacén de políticas

Puede eliminar un almacén de políticas mediante la `delete-policy-store` operación y *`PSEXAMPLEabcdefg111111`* sustituirlo por el ID del almacén de políticas que desee eliminar.

```
$ aws verifiedpermissions delete-policy-store \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Si se ejecuta correctamente, este comando no genera ninguna salida.

> **Note**
>
> Si la protección contra la eliminación está habilitada para este almacén de políticas, primero debe ejecutar la `update-policy-store` operación y deshabilitar la protección contra la eliminación.
>
> ```
> aws verifiedpermissions update-policy-store \
> ```

```
    --deletion-protection "DISABLED" \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Alias del almacén de políticas de permisos verificados de Amazon

Un alias de almacén de políticas es un nombre descriptivo para un almacén de políticas. Por ejemplo, los alias del almacén de políticas permiten hacer referencia a un almacén de políticas utilizando `policy-store-alias/example-policy-store` en lugar de`PSEXAMPLEabcdefg111111`. Los alias del almacén de políticas se pueden usar en cualquier operación de permisos verificados que acepte un parámetro `policyStoreId` de entrada.

Puede crear un alias de almacén de políticas para un almacén de políticas mediante la `CreatePolicyStoreAlias` API o mediante el `AWS::VerifiedPermissions::PolicyStoreAlias` CloudFormation recurso.

La API de permisos verificados de Amazon proporciona un control total de los alias de los almacenes de políticas en cada Cuenta de AWS región. La API incluye operaciones para crear un alias del almacén de políticas (`CreatePolicyStoreAlias`), ver los nombres de los alias del almacén de políticas y los ARN de los alias del almacén de políticas (`GetPolicyStoreAlias`,`ListPolicyStoreAliases`) y eliminar un alias del almacén de políticas (`DeletePolicyStoreAlias`).

Temas

- Propiedades de los alias de los almacenes de políticas
- Creación de alias de almacén de políticas de permisos verificados de Amazon
- Recuperación de los alias de la tienda de políticas de permisos verificados de Amazon
- Eliminar los alias de la tienda de políticas de permisos verificados de Amazon
- Uso de alias de almacenamiento de la política de permisos verificados de Amazon en las operaciones de la API
- Control del acceso a los alias del almacén de políticas

Propiedades de los alias de los almacenes de políticas

Cómo funcionan los alias de las tiendas de políticas en Amazon Verified Permissions.

El alias de un almacén de políticas es independiente AWS recurso

El alias de un almacén de políticas no es propiedad de un almacén de políticas. Las acciones que realice con el alias del almacén de políticas no afectan al almacén de políticas asociado. Puede eliminar el alias del almacén de políticas sin que ello afecte al almacén de políticas asociado.
Si elimina un almacén de políticas, también se eliminan todos los alias del almacén de políticas asociados a ese almacén de políticas.

Cada alias del almacén de políticas tiene un nombre de recurso de Amazon (ARN) que identifica de forma exclusiva el alias del almacén de políticas. Si especifica un alias de almacén de políticas como recurso en una política de IAM, la política se refiere al alias del almacén de políticas, no al almacén de políticas asociado.

Cada alias del almacén de políticas tiene dos formatos

Al crear un alias del almacén de políticas, se especifica el nombre del alias del almacén de políticas. Amazon Verified Permissions crea automáticamente el alias ARN del almacén de políticas.

- El ARN de un alias del almacén de políticas es un nombre de recurso de Amazon (ARN) que identifica de forma exclusiva el alias del almacén de políticas.

```
# Alias ARN
arn:aws:verifiedpermissions:us-east-1:123456789012:policy-store-alias/example-policy-store
```

- Un nombre de alias del almacén de políticas que es único en la región Cuenta de AWS and. En la API de permisos verificados de Amazon, el nombre del alias del almacén de políticas siempre lleva el prefijo. `policy-store-alias/`

```
# Alias name
policy-store-alias/example-policy-store
```

Los alias de los almacenes de políticas no son secretos

Los alias del almacén de políticas pueden mostrarse en texto plano en CloudTrail los registros y otros resultados. No incluya información confidencial o delicada en el nombre del alias del almacén de políticas.

Cada alias del almacén de políticas está asociado a un almacén de políticas a la vez

El alias del almacén de políticas y el almacén de políticas asociado deben pertenecer a la misma Cuenta de AWS región. Puede asociar un alias de almacén de políticas a cualquier almacén de políticas de la misma Cuenta de AWS región.

Por ejemplo, este `ListPolicyStoreAliases` resultado muestra que el alias del almacén de `example-policy-store` políticas está asociado exactamente a un almacén de políticas de destino, que se representa mediante la `policyStoreId` propiedad.

```
{
    "aliasName": "policy-store-alias/example-policy-store",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-store-alias/
example-policy-store",
    "createdAt": "2024-01-15T12:30:00.000000+00:00",
    "state": "Active"
}
```

Se pueden asociar varios alias al mismo almacén de políticas

Por ejemplo, puede asociar los `example-policy-store-2` alias `example-policy-store` y al mismo almacén de políticas.

```
[
    {
        "aliasName": "policy-store-alias/example-policy-store",
        "policyStoreId": "PSEXAMPLEabcdefg111111",
        "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-store-
alias/example-policy-store",
        "createdAt": "2024-01-15T12:30:00.000000+00:00",
        "state": "Active"
    },
    {
        "aliasName": "policy-store-alias/example-policy-store-2",
        "policyStoreId": "PSEXAMPLEabcdefg111111",
        "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-store-
alias/example-policy-store-2",
        "createdAt": "2024-01-16T09:15:00.000000+00:00",
        "state": "Active"
    }
]
```

El alias de un almacén de políticas debe ser único en un Cuenta de AWS y región

Por ejemplo, solo puede tener un alias de almacén de políticas con el nombre `example-policy-store` en cada Cuenta de AWS región. Los alias de los almacenes de políticas distinguen entre mayúsculas y minúsculas. No puede cambiar el nombre de alias de un almacén de políticas. Sin embargo, puede eliminar el alias del almacén de políticas y crear un nuevo alias del almacén de políticas con el nombre deseado una vez transcurrido el período de reserva de 24 horas.

Puede crear alias de almacén de políticas con el mismo nombre en distintas regiones. Cada alias de almacén de políticas tendrá un ARN único. Si su código hace referencia a un nombre de alias de almacén de políticas`policy-store-alias/example-policy-store`, puede ejecutarlo en varias regiones. En cada región, utiliza un almacén de políticas diferente.

Los alias del almacén de políticas admiten eliminaciones temporales y definitivas

De forma predeterminada, cuando se elimina un alias del almacén de políticas, se elimina temporalmente. El nombre del alias del almacén de políticas se reserva durante un período de 24 horas. Si intenta crear un alias de almacén de políticas con el mismo nombre durante este período, se rechazará la solicitud. Durante este período, `GetPolicyStoreAlias` devuelve el alias del almacén de políticas con el `PendingDeletion` estado. También puede eliminar por completo un alias de almacén de políticas `HardDelete` especificándolo como`deletionMode`. Un alias del almacén de políticas eliminado de forma permanente se elimina inmediatamente y el nombre del alias del almacén de políticas queda disponible para su reutilización inmediata. Para obtener más información, consulte Eliminar los alias de la tienda de políticas de permisos verificados de Amazon.

Puede usar alias para identificar los almacenes de políticas

Puede usar un alias de almacén de políticas para identificar un almacén de políticas en todas las operaciones que acepten un `policyStoreId` (por ejemplo,`IsAuthorized`). En esos casos, el nombre del alias del almacén de políticas debe ir precedido de`policy-store-alias/`. Los alias del almacén de políticas no se pueden utilizar para identificar un almacén de políticas para la `DeletePolicyStore` operación.

No puede utilizar un nombre de alias de almacén de políticas o un ARN de alias de almacén de políticas para identificar un almacén de políticas en el `Resource` elemento de una política de IAM. Para controlar el acceso a un almacén de políticas cuando se hace referencia a él mediante un alias de almacén de políticas, consulte. Control del acceso a los alias del almacén de políticas

Creación de alias de almacén de políticas de permisos verificados de Amazon

Puede crear un alias de almacén de políticas para hacer referencia a un almacén de políticas mediante un nombre descriptivo. El nombre del alias de un almacén de políticas debe ser único por Cuenta de AWS región. Los alias del almacén de políticas solo se pueden asociar a los almacenes de políticas que sean propiedad del mismo propietario Cuenta de AWS y estén activos en la misma región que el alias del almacén de políticas. Los alias de los almacenes de políticas son recursos independientes con su propia autorización ARNs y la de IAM.

De forma predeterminada, solo se pueden asociar 10 alias del almacén de políticas al mismo almacén de políticas.

> **Note**
>
> `CreatePolicyStoreAliases` idempotente. Si llama a la `CreatePolicyStoreAlias` operación con un nombre de alias del almacén de políticas y un identificador del almacén de políticas que coinciden con un alias del almacén de políticas existente, la `CreatePolicyStoreAlias` operación se realiza correctamente y devuelve el alias del almacén de políticas existente. Sin embargo, si llama a la `CreatePolicyStoreAlias` operación con un nombre de alias del almacén de políticas existente pero con un ID de almacén de políticas diferente, la operación devuelve un`ConflictException`.

AWS CLI

Para crear un alias de almacén de políticas

Puede crear un alias de almacén de políticas mediante la CreatePolicyStoreAliasoperación. En el siguiente ejemplo, se crea un alias de almacén de políticas con ese nombre`example-policy-store`.

```
$ aws verifiedpermissions create-policy-store-alias \
    --alias-name policy-store-alias/example-policy-store \
    --policy-store-id PSEXAMPLEabcdefg111111
{
    "aliasName": "policy-store-alias/example-policy-store",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
```

```
    "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-store-
alias/example-policy-store",
    "createdAt": "2024-01-15T12:30:00.000000+00:00"
}
```

Recuperación de los alias de la tienda de políticas de permisos verificados de Amazon

Puede recuperar información sobre los alias del almacén de políticas mediante la `GetPolicyStoreAlias` operación para obtener detalles sobre un alias de almacén de políticas específico o la `ListPolicyStoreAliases` operación para enumerar todos los alias del almacén de políticas de su región y de su Cuenta de AWS región.

Obtener un alias de almacén de políticas

Utilice la `GetPolicyStoreAlias` operación para recuperar los detalles sobre un alias de almacén de políticas específico, incluido el ID del almacén de políticas asociado.

AWS CLI

Para recuperar detalles sobre un alias de almacén de políticas

Puede recuperar un alias de almacén de políticas mediante la GetPolicyStoreAliasoperación. En el siguiente ejemplo, se recuperan los detalles de un alias de almacén de políticas con el nombre`example-policy-store`.

```
$ aws verifiedpermissions get-policy-store-alias \
    --alias-name policy-store-alias/example-policy-store
{
    "aliasName": "policy-store-alias/example-policy-store",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-store-
alias/example-policy-store",
    "createdAt": "2024-01-15T12:30:00.000000+00:00",
    "state": "Active"
}
```

Listar los alias de las tiendas de políticas

Utilice la `ListPolicyStoreAliases` operación para enumerar todos los alias del almacén de políticas de su región Cuenta de AWS . Puede usar el `filter` parámetro para enumerar solo los alias del almacén de políticas asociados a un almacén de políticas específico.

AWS CLI

Para enumerar todos los alias de los almacenes de políticas

Puede enumerar los alias de los almacenes de políticas mediante la ListPolicyStoreAliasesoperación. En el siguiente ejemplo, se enumeran todos los alias del almacén de políticas propiedad del 123456789012 Cuenta de AWS en la región us-west-2.

```
$ aws verifiedpermissions list-policy-store-aliases
{
    "policyStoreAliases": [
        {
            "aliasName": "policy-store-alias/example-policy-store",
            "policyStoreId": "PSEXAMPLEabcdefg111111",
            "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-
store-alias/example-policy-store",
            "createdAt": "2024-01-15T12:30:00.000000+00:00",
            "state": "Active"
        },
        {
            "aliasName": "policy-store-alias/example-policy-store-2",
            "policyStoreId": "PSEXAMPLEabcdefg111111",
            "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-
store-alias/example-policy-store-2",
            "createdAt": "2024-01-16T09:15:00.000000+00:00",
            "state": "Active"
        },
        {
            "aliasName": "policy-store-alias/example-policy-store-3",
            "policyStoreId": "PSEXAMPLEabcdefg222222",
            "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-
store-alias/example-policy-store-3",
            "createdAt": "2024-01-17T14:45:00.000000+00:00",
            "state": "Active"
        }
    ]
```

```
}
```

Para enumerar los alias del almacén de políticas de un almacén de políticas específico

Utilice el `filter` parámetro para enumerar únicamente los alias asociados a un almacén de políticas específico.

```
$ aws verifiedpermissions list-policy-store-aliases \
    --filter '{"policyStoreId": "PSEXAMPLEabcdefg111111"}'
{
    "policyStoreAliases": [
        {
            "aliasName": "policy-store-alias/example-policy-store",
            "policyStoreId": "PSEXAMPLEabcdefg111111",
            "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-
store-alias/example-policy-store",
            "createdAt": "2024-01-15T12:30:00.000000+00:00",
            "state": "Active"
        },
        {
            "aliasName": "policy-store-alias/example-policy-store-2",
            "policyStoreId": "PSEXAMPLEabcdefg111111",
            "aliasArn": "arn:aws:verifiedpermissions:us-west-2:123456789012:policy-
store-alias/example-policy-store-2",
            "createdAt": "2024-01-16T09:15:00.000000+00:00",
            "state": "Active"
        }
    ]
}
```

Eliminar los alias de la tienda de políticas de permisos verificados de Amazon

Puede eliminar un alias de almacén de políticas cuando ya no lo necesite. La eliminación de un alias de almacén de políticas no afecta al almacén de políticas asociado. Al eliminar un almacén de políticas, se eliminan todos los alias del almacén de políticas asociados a ese almacén de políticas.

Amazon Verified Permissions admite dos modos de eliminación para los alias de los almacenes de políticas:

- Eliminación temporal (predeterminada): el alias del almacén de políticas pasa a un `PendingDeletion` estado. El nombre del alias del almacén de políticas está reservado durante 24 horas y no se puede volver a utilizar durante este período. Durante este período, `GetPolicyStoreAlias` devuelve el alias del almacén de políticas con el `PendingDeletion` estado. Este es el comportamiento predeterminado cuando no se especifica un `deletionMode` o cuando se especifica`SoftDelete`.
- Eliminación definitiva: el alias del almacén de políticas se elimina inmediatamente. El nombre del alias del almacén de políticas está disponible para su reutilización inmediata. Para realizar una eliminación definitiva, `HardDelete` especifíquelo como`deletionMode`.

Eliminación gradual de un alias de almacén de políticas

De forma predeterminada, al eliminar un alias del almacén de políticas se realiza una eliminación temporal. El alias del almacén de políticas introduce el `PendingDeletion` estado y el nombre del alias del almacén de políticas se reserva durante 24 horas.

AWS CLI

Para eliminar temporalmente un alias del almacén de políticas

Puede eliminar automáticamente un alias de almacén de políticas mediante la DeletePolicyStoreAliasoperación. En el siguiente ejemplo, se elimina temporalmente un alias de almacén de políticas con ese nombre`example-policy-store`.

```
$ aws verifiedpermissions delete-policy-store-alias \
    --alias-name policy-store-alias/example-policy-store
```

También puede especificar de forma explícita el modo de borrado temporal.

```
$ aws verifiedpermissions delete-policy-store-alias \
    --alias-name policy-store-alias/example-policy-store \
    --deletion-mode SoftDelete
```

Eliminar de forma difícil un alias de almacén de políticas

Para eliminar inmediatamente un alias de almacén de políticas, `HardDelete` especifíquelo como`deletionMode`. Un alias del almacén de políticas eliminado de forma permanente no introduce

el `PendingDeletion` estado y el nombre del alias del almacén de políticas queda disponible para su reutilización inmediata.

Si elimina de forma definitiva un alias del almacén de políticas que anteriormente se había eliminado de forma temporal, el alias del almacén de políticas se eliminará inmediatamente.

> **Important**
>
> Los permisos verificados de Amazon finalmente son consistentes. Si eliminas por completo el alias de un almacén de políticas y lo vuelves a crear inmediatamente para que apunte a un almacén de políticas diferente, las solicitudes que hacen referencia al alias del almacén de políticas pueden seguir resolviéndose en el almacén de políticas anteriormente asociado durante un breve período de tiempo. Para evitar que la autorización se produzca de forma inesperada, espere un tiempo para que la eliminación se propague antes de volver a crear un alias de almacén de políticas con el mismo nombre.

AWS CLI

Para eliminar por completo un alias de almacén de políticas

Puede eliminar por completo un alias de un almacén de políticas mediante la DeletePolicyStoreAliasoperación con el `deletion-mode` parámetro establecido en`HardDelete`. En el siguiente ejemplo, se elimina inmediatamente un alias del almacén de políticas con ese nombre`example-policy-store`.

```
$ aws verifiedpermissions delete-policy-store-alias \
    --alias-name policy-store-alias/example-policy-store \
    --deletion-mode HardDelete
```

Uso de alias de almacenamiento de la política de permisos verificados de Amazon en las operaciones de la API

Cualquier operación de permisos verificados de Amazon que acepte un `policyStoreId` parámetro, como IsAuthorized, y IsAuthorizedWithToken GetPolicyStore, puede aceptar el nombre de un alias de la tienda de políticas en lugar del ID de la tienda de políticas.

> **Important**
>
> Cuando utilices un alias de un almacén de políticas como valor de un `policyStoreId` parámetro, debes incluir el `policy-store-alias/` prefijo. Por ejemplo, use`policy-store-alias/example-policy-store`, no`example-policy-store`.

Uso de los alias del almacén de políticas en Operations

El siguiente `IsAuthorized` comando usa un alias de almacén de políticas con el nombre `example-policy-store` para identificar un almacén de políticas.

AWS CLI

```
$ aws verifiedpermissions is-authorized \
    --policy-store-id policy-store-alias/example-policy-store \
    --principal entityType=User,entityId=alice \
    --action actionType=Action,actionId=view \
    --resource entityType=Photo,entityId=photo123
```

> **Note**
>
> No puede usar un alias de almacén de políticas en lugar del `policyStoreId` campo de la DeletePolicyStore operación.

Uso de los alias del almacén de políticas en Across Regiones de AWS

Uno de los usos más potentes de los alias es en aplicaciones que se ejecutan en múltiples Regiones de AWS. Por ejemplo, es posible que tenga una aplicación global que utilice diferentes almacenes de políticas en cada región.

- En us-east-1, quieres usar. `PSEXAMPLEabcdefg111111`
- En eu-west-1, quieres usar. `PSEXAMPLEabcdefg222222`

Puedes crear una versión diferente de tu aplicación en cada región o usar un diccionario o una sentencia switch para seleccionar el almacén de políticas adecuado para cada región. Sin embargo,

es mucho más fácil crear un alias de almacén de políticas con el mismo nombre de alias de almacén de políticas en cada región. Recuerde que el nombre del alias del almacén de políticas distingue entre mayúsculas y minúsculas.

AWS CLI

```
$ aws --region us-east-1 verifiedpermissions create-policy-store-alias \
    --alias-name policy-store-alias/my-app \
    --policy-store-id PSEXAMPLEabcdefg111111

$ aws --region eu-west-1 verifiedpermissions create-policy-store-alias \
    --alias-name policy-store-alias/my-app \
    --policy-store-id PSEXAMPLEabcdefg222222
```

Luego, usa el alias del almacén de políticas en tu código. Cuando tu código se ejecute en cada región, el alias del almacén de políticas hará referencia al almacén de políticas asociado en esa región.

AWS CLI

```
$ aws verifiedpermissions is-authorized \
    --policy-store-id policy-store-alias/my-app \
    --principal entityType=User,entityId=alice \
    --action actionType=Action,actionId=view \
    --resource entityType=Photo,entityId=photo123
```

Sin embargo, existe el riesgo de que se elimine el alias del almacén de políticas. En ese caso, los intentos de la aplicación de usar el nombre del alias del almacén de políticas fallarán y es posible que tengas que volver a crear o actualizar el alias del almacén de políticas. Para mitigar este riesgo, ten cuidado a la hora de dar permiso a los directores para que administren los alias del almacén de políticas que utilizas en tu aplicación.

Los alias del almacén de políticas no son un mecanismo de control del tráfico

Un alias de almacén de políticas es un nombre estable y fácil de usar para un almacén de políticas. No es un mecanismo para desplazar, dividir o ponderar el tráfico de autorización entre los almacenes de políticas. Si está familiarizado con las funciones que dirigen el tráfico entre las versiones de un

recurso, como los alias de Lambda, tenga en cuenta que los alias de los almacenes de políticas no se comportan de la misma manera. Por diseño, un alias de almacén de políticas se parece más a un alias de AWS KMS clave: proporciona un nombre duradero para un recurso, no una capa de enrutamiento delante de él.

Por este motivo, no proporcionamos una `UpdatePolicyStoreAlias` operación de forma intencionada. Para cambiar el almacén de políticas al que apunta el alias de un almacén de políticas, debes eliminar el alias del almacén de políticas y crear un nuevo alias de almacén de políticas que tenga el mismo nombre y se dirija a un almacén de políticas diferente. No se trata de una operación atómica y no ofrece las garantías que ofrecería un mecanismo de control del tráfico.

Cuando cambias el almacén de políticas en el que se resuelve el alias de un almacén de políticas, el cambio no se aplica en todas partes al mismo tiempo:

- La asignación actualizada debe propagarse desde el plano de control al plano de datos que evalúa las solicitudes de autorización. La asignación no se propaga de forma instantánea.
- Como la resolución de los alias de los almacenes de políticas es, en última instancia, coherente y se puede almacenar en caché, existe una ventana de transición después de un cambio. Durante este período, el almacén de políticas asociado anteriormente atiende algunas solicitudes y el almacén de políticas recién asociado atiende otras solicitudes. La duración de esta ventana es indeterminada.

Durante este período, su aplicación puede recibir las decisiones de autorización de cualquiera de los almacenes de políticas. Como los distintos almacenes de políticas pueden contener políticas, entidades y esquemas diferentes, la misma solicitud puede generar decisiones diferentes según el almacén de políticas que la reciba. Por lo tanto, los alias de los almacenes de políticas no pueden proporcionar una separación atómica entre los almacenes de políticas y no sustituyen a una estrategia de implementación o administración del tráfico.

⚠ No utilices los alias como mecanismo de transición

No cree una infraestructura como código de nivel superior ni una primitiva de implementación que redirija el alias de un almacén de políticas para cambiar el tráfico de autorización de un almacén de políticas a otro. Como el cambio no es atómico, las solicitudes se pueden autorizar en ambos almacenes de políticas al mismo tiempo durante un período indeterminado. Esto puede llevar a decisiones de autorización incoherentes. Para cambiar de

almacén de políticas de forma segura, consulteRealizar cambios en el almacén de políticas sin tiempo de inactividad.

Realizar cambios en el almacén de políticas sin tiempo de inactividad

Como los alias de los almacenes de políticas no proporcionan una separación atómica (consulteLos alias del almacén de políticas no son un mecanismo de control del tráfico), le recomendamos que evite migrar de un almacén de políticas a otro redirigiendo el alias del almacén de políticas. En su lugar, controla la migración desde tu aplicación para poder validar el nuevo almacén de políticas y revertirlo al instante si es necesario. El siguiente enfoque le permite cambiar de almacén de políticas sin tiempo de inactividad:

1. Cree el nuevo almacén de políticas y asígnele su propio alias de almacén de políticas, de modo que el almacén de políticas actual y el nuevo almacén de políticas tengan cada uno un nombre distinto y estable. Por ejemplo, siga `policy-store-alias/example-policy-store` apuntando a su almacén de políticas actual y `policy-store-alias/example-policy-store-2` créelo para el nuevo almacén de políticas. No reutilices ni cambies el alias de un único almacén de políticas para realizar el cambio.
2. Replique sus políticas, esquemas y cualquier otra configuración en el nuevo almacén de políticas y ejecútelo en paralelo con el almacén de políticas actual.
3. Agregue un valor de configuración o un indicador de función a su aplicación (por ejemplo, mediante el uso de AppConfig) que determine qué alias del almacén de políticas es el más autorizado para las decisiones de autorización. No confíes en cambiar el alias en sí mismo para cambiar el tráfico.
4. Ejecute el nuevo almacén de políticas en modo oculto. Envíe el mismo almacén de políticas `IsAuthorized` o `IsAuthorizedWithToken` una solicitud a ambos almacenes de políticas y compare las decisiones. Registra e investiga cualquier discrepancia hasta que el nuevo almacén de políticas te dé las decisiones que esperabas.
5. Usa el indicador de función para cambiar las decisiones de autorización al nuevo alias del almacén de políticas de forma gradual, por ejemplo, mediante una implementación gradual en todos los hosts de aplicaciones o segmentos de usuarios. Supervisa las decisiones de autorización y las tasas de error a medida que avanzas.
6. Usa el indicador de función para volver inmediatamente al alias original del almacén de políticas si detectas un problema. Como la aplicación controla el cambio, la reversión es instantánea y no depende de la propagación del alias.

7. Elimine el antiguo almacén de políticas y su alias de almacén de políticas una vez que el nuevo almacén de políticas esté completamente validado y sirva a todo el tráfico.

Este patrón le da a la aplicación, en lugar de a la propagación de alias, el control sobre qué almacén de políticas atiende cada solicitud. Ese control es lo que hace que la migración sea segura y reversible, y evita la ventana de transición que supondría cambiar el alias de un almacén de políticas.

Control del acceso a los alias del almacén de políticas

Los responsables que administran los alias del almacén de políticas deben tener permiso para interactuar con esos alias del almacén de políticas y, en algunas operaciones, con el almacén de políticas al que está asociado el alias del almacén de políticas. Puede proporcionar estos permisos mediante políticas. IAM

En las siguientes secciones se describen los permisos necesarios para crear y administrar los alias de los almacenes de políticas.

permisos verificados: CreatePolicyStoreAlias

Para crear un alias de almacén de políticas, el director necesita los siguientes permisos tanto para el alias del almacén de políticas como para el almacén de políticas asociado.

- `verifiedpermissions:CreatePolicyStoreAlias`para el alias del almacén de políticas. Proporcione este permiso en una IAM política adjunta al director, que puede crear el alias del almacén de políticas.

  El siguiente ejemplo de declaración de política especifica un alias de almacén de políticas concreto en un `Resource` elemento. Sin embargo, puede enumerar varios alias de almacenes de políticas ARNs o especificar un patrón de alias de almacenes de políticas, como`"sample*"`. También puede especificar un `Resource` valor de `"*"` para que la entidad principal pueda crear cualquier alias de almacén de políticas en la región Cuenta de AWS y.

  ```
  {
    "Sid": "IAMPolicyForCreateAlias",
    "Effect": "Allow",
    "Action": "verifiedpermissions:CreatePolicyStoreAlias",
    "Resource": "arn:aws:verifiedpermissions:us-east-1:123456789012:policy-store-alias/
  example-policy-store"
  }
  ```

- `verifiedpermissions:CreatePolicyStoreAlias`para el almacén de políticas asociado. Este permiso debe proporcionarse en una IAM política.

```
{
  "Sid": "PolicyStorePermissionForAlias",
  "Effect": "Allow",
  "Action": "verifiedpermissions:CreatePolicyStoreAlias",
  "Resource": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
}
```

permisos verificados: GetPolicyStoreAlias

Para obtener detalles sobre un alias de almacén de políticas específico, el director debe tener `verifiedpermissions:GetPolicyStoreAlias` permiso para usar el alias del almacén de políticas en una IAM política.

El siguiente ejemplo de declaración de política otorga al principal permiso para obtener un alias de almacén de políticas específico.

```
{
  "Sid": "IAMPolicyForGetAlias",
  "Effect": "Allow",
  "Action": "verifiedpermissions:GetPolicyStoreAlias",
  "Resource": "arn:aws:verifiedpermissions:us-east-1:123456789012:policy-store-alias/
example-policy-store"
}
```

permisos verificados: ListPolicyStoreAliases

Para incluir los alias de los almacenes de políticas en la región Cuenta de AWS y, el director debe tener `verifiedpermissions:ListPolicyStoreAliases` permiso en una política. IAM Como esta política no está relacionada con ningún almacén de políticas o recurso de alias de almacén de políticas en particular, el valor del elemento de recurso de la política debe serlo"`*`".

Por ejemplo, la siguiente declaración IAM de política otorga al director el permiso para enumerar todos los alias de los almacenes de políticas del Cuenta de AWS.

```
{
  "Sid": "IAMPolicyForListingAliases",
```

```
  "Effect": "Allow",
  "Action": "verifiedpermissions:ListPolicyStoreAliases",
  "Resource": "*"
}
```

permisos verificados: DeletePolicyStoreAlias

Para eliminar un alias del almacén de políticas, el director solo necesita permiso para el alias del almacén de políticas.

> **Note**
>
> La eliminación de un alias del almacén de políticas no afecta al almacén de políticas asociado, aunque las aplicaciones que hagan referencia al alias del almacén de políticas recibirán errores. Si elimina por error un alias de almacén de políticas, puede volver a crearlo después del período de reserva de 24 horas.

El director necesita `verifiedpermissions:DeletePolicyStoreAlias` permiso para el alias del almacén de políticas. Proporcione este permiso en una IAM política adjunta al director, quien podrá eliminar el alias del almacén de políticas.

El siguiente ejemplo de declaración de política especifica el alias del almacén de políticas en un `Resource` elemento. Sin embargo, puede enumerar varios alias de almacenes de políticas ARNs o especificar un patrón de alias de almacenes de políticas, como`"sample*"`. También puede especificar un `Resource` valor de `"*"` para permitir que la entidad principal elimine cualquier alias de almacén de políticas de la región Cuenta de AWS y.

```
{
  "Sid": "IAMPolicyForDeleteAlias",
  "Effect": "Allow",
  "Action": "verifiedpermissions:DeletePolicyStoreAlias",
  "Resource": "arn:aws:verifiedpermissions:us-east-1:123456789012:policy-store-alias/
example-policy-store"
}
```

Limitar los permisos del alias del almacén de políticas

Puede usar un alias de almacén de políticas para hacer referencia a un almacén de políticas en cualquier operación que acepte un `policyStoreId` campo como entrada. Al hacerlo, Amazon

Verified Permissions autoriza `verifiedpermissions:GetPolicyStoreAlias` contra el alias del almacén de políticas y la operación solicitada contra el almacén de políticas asociado.

Por ejemplo, si la `IsAuthorized` operación se realiza con un alias de almacén de políticas, el director necesita ambas cosas:

- `verifiedpermissions:GetPolicyStoreAlias`permiso para el alias del almacén de políticas
- `verifiedpermissions:IsAuthorized`permiso para el almacén de políticas asociado

El siguiente ejemplo de política otorga permiso para llamar `IsAuthorized` con un alias de almacén de políticas específico.

```
{
  "Sid": "IAMPolicyForAliasUsage",
  "Effect": "Allow",
  "Action": "verifiedpermissions:GetPolicyStoreAlias",
  "Resource": "arn:aws:verifiedpermissions:us-east-1:123456789012:policy-store-alias/
example-policy-store"
},
{
  "Sid": "IAMPolicyForPolicyStoreOperation",
  "Effect": "Allow",
  "Action": "verifiedpermissions:IsAuthorized",
  "Resource": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
}
```

Para limitar los alias del almacén de políticas que puede usar un director, restrinja el `verifiedpermissions:GetPolicyStoreAlias` permiso. Por ejemplo, la siguiente política permite al director utilizar cualquier alias del almacén de políticas, excepto los que comiencen por. `Restricted`

```
{
  "Sid": "IAMPolicyForAliasAllow",
  "Effect": "Allow",
  "Action": "verifiedpermissions:GetPolicyStoreAlias",
  "Resource": "arn:aws:verifiedpermissions:us-east-1:123456789012:policy-store-alias/*"
},
{
  "Sid": "IAMPolicyForAliasDeny",
  "Effect": "Deny",
```

```
  "Action": "verifiedpermissions:GetPolicyStoreAlias",
  "Resource": "arn:aws:verifiedpermissions:us-east-1:123456789012:policy-store-alias/
Restricted*"
}
```

Esquema del almacén de políticas de Amazon Verified Permissions.

Un esquema es una declaración de la estructura de los tipos de entidad que admite la aplicación y de las acciones que la aplicación puede proporcionar en las solicitudes de autorización. Para ver la diferencia entre la forma en que Verified Permissions y Cedar gestionan los esquemas, consulta Compatibilidad con esquemas.

Para obtener información más detallada consulte Formato de esquemas Cedar en la Guía de referencia sobre el lenguaje de las políticas de Cedar.

> **Note**
>
> El uso de esquemas en Verified Permissions es opcional, pero se recomienda encarecidamente su uso para el software de producción. Cuando se crea una nueva política, Verified Permissions puede usar el esquema para validar las entidades y los atributos a los que se hace referencia en el ámbito y las condiciones a fin de evitar errores tipográficos y errores en las políticas que puedan provocar un comportamiento errático del sistema. Si activa la validación de políticas, todas las políticas nuevas deben ajustarse al esquema.

Consola de administración de AWS

Para crear un esquema

1. Abre el Consola de permisos verificados. Elige tu tienda de pólizas.
2. En el panel de navegación de la izquierda, seleccione Esquema.
3. Elija Create schema (Crear esquema).

AWS CLI

Para enviar un esquema nuevo o sobrescribir un esquema existente mediante el AWS CLI.

Puede crear un almacén de políticas ejecutando un AWS CLI comando similar al siguiente ejemplo.

Considere un esquema que contenga el siguiente contenido de Cedar:

```
{
    "MySampleNamespace": {
        "actions": {
            "remoteAccess": {
                "appliesTo": {
                    "principalTypes": [ "Employee" ]
                }
            }
        },
        "entityTypes": {
            "Employee": {
                "shape": {
                    "type": "Record",
                    "attributes": {
                        "jobLevel": {"type": "Long"},
                        "name": {"type": "String"}
                    }
                }
            }
        }
    }
}
```

Primero debe convertir el JSON en una cadena de una sola línea y comenzar con una instrucción de su tipo de datos: `cedarJson`. El siguiente ejemplo utiliza el siguiente contenido del archivo `schema.json` que incluye la versión con el carácter de escape del esquema JSON.

> **Note**
>
> El ejemplo se muestra con ajustes de línea para facilitar la lectura. Debe tener todo el archivo en una sola línea para que el comando lo acepte.

```
{"cedarJson": "{\"MySampleNamespace\": {\"actions\": {\"remoteAccess\": {\"appliesTo
\":
{\"principalTypes\": [\"Employee\"]}}},\"entityTypes\": {\"Employee\": {\"shape\":
{\"attributes\": {\"jobLevel\": {\"type\": \"Long\"},\"name\": {\"type\": \"String
\"}},
\"type\": \"Record\"}}}}}"}
```

```
$ aws verifiedpermissions put-schema \
```

```
    --definition file://schema.json \
    --policy-store PSEXAMPLEabcdefg111111
{
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "namespaces": [
        "MySampleNamespace"
    ],
    "createdDate": "2023-07-17T21:07:43.659196+00:00",
    "lastUpdatedDate": "2023-08-16T17:03:53.081839+00:00"
}
```

AWS SDKs

Puede crear un almacén de políticas mediante la API `PutSchema`. Para obtener más información, consulte PutSchema en la Guía de referencia de la API de permisos verificados de Amazon.

Edición de esquemas de almacenamiento de políticas

Cuando seleccionas Esquema en la consola de permisos verificados de Amazon, Tipos de entidades y Acciones se muestran las que componen su esquema. Puedes ver y editar tu esquema en cualquiera de las dos Modo visual o modo JSON. El modo visual permite actualizar el esquema añadiendo nuevos tipos y acciones mediante varios asistentes. Con el modo JSON, puedes empezar a actualizar el código JSON del esquema directamente en el editor JSON.

Visual Mode

El editor visual de esquemas comienza con una serie de diagramas que ilustran las relaciones entre las entidades de tu esquema. Elija Expandir para maximizar la visualización de los diagramas. Hay dos diagramas disponibles:

- Diagrama de acciones — El Diagrama de acciones la vista muestra los tipos de Directores ha configurado en su almacén de políticas, el Acciones son elegibles para desempeñarse, y el Recursos sobre los que son aptos para realizar acciones. Las líneas entre las entidades indican su capacidad para crear una política que permita al principal tomar medidas sobre un recurso. Si tu diagrama de acciones no indica una relación entre dos entidades, debes crear esa relación entre ellas para poder permitirla o denegarla en las políticas. Selecciona una entidad para ver un resumen de las propiedades y profundiza para ver todos los detalles. Elige Filtrar por esto [acción | tipo de recurso | tipo de principal] para ver una entidad en una vista con solo sus propias conexiones.

- Diagrama de tipos de entidades — El Diagrama de tipos de entidades se centra en las relaciones entre los directores y los recursos. Si desea comprender las complejas relaciones entre los padres anidados de su esquema, revise este diagrama. Pase el ratón por encima de una entidad para ver en detalle las relaciones principales que tiene.

  Debajo de los diagramas hay vistas de lista de Tipos de entidades y Acciones en tu esquema. La vista de lista es útil cuando quieres ver inmediatamente los detalles de un tipo de acción o entidad específico. Seleccione cualquier entidad para ver los detalles.

Para editar un esquema de Verified Permissions en modo visual

1. Abra el Consola de permisos verificados. Elige tu tienda de pólizas.
2. En el panel de navegación de la izquierda, seleccione Esquema.
3. Elija Modo visual. Revisa los diagramas de entidad-relación y planifica los cambios que deseas realizar en tu esquema. Si lo desea, puede Filtrar por una entidad para examinar sus conexiones individuales con otras entidades.
4. Elija Edit schema (Editar esquema).
5. En la sección Detalles, escriba un espacio de nombres para su esquema.
6. En la sección Tipos de entidad, seleccione Agregar nuevo tipo de entidad.
7. Escriba el nombre de la entidad.
8. (Opcional) Seleccione Agregar un elemento principal para añadir las entidades principales a las que pertenece la nueva entidad. Para eliminar un elemento principal que se haya agregado a la entidad, seleccione Eliminar junto al nombre del elemento principal.
9. Para añadir otro atributo, seleccione Agregar un atributo. Escriba el nombre del atributo y elija el tipo de atributo para cada atributo de la entidad. Verified Permissions utiliza los valores de atributo especificados al verificar las políticas con el esquema. Seleccione si cada atributo es obligatorio. Para eliminar un atributo que se ha añadido a la entidad, seleccione Eliminar junto al atributo.
10. Seleccione Agregar tipo de entidad para añadir la entidad al esquema.
11. En la sección Acciones, seleccione Agregar acción nueva.
12. Escriba el nombre de la acción.
13. (Opcional) Seleccione Agregar un recurso para añadir los tipos de recurso a los que se refiere la acción. Para eliminar un tipo de recurso que se ha agregado a la acción, seleccione Eliminar junto al nombre del tipo de recurso.

14. (Opcional) Seleccione Agregar una entidad principal para añadir el tipo de entidad principal al que se refiere la acción. Para eliminar un tipo de entidad principal que se ha agregado a la acción, seleccione Eliminar junto al nombre del tipo de entidad principal.
15. Elija Añadir un atributo para agregar atributos que se pueden agregar al contexto de una acción en sus solicitudes de autorización. Introduzca el Nombre del atributo y elija el Tipo de atributo para cada atributo. Verified Permissions utiliza los valores de atributo especificados al verificar las políticas con el esquema. Seleccione si cada atributo es obligatorio. Para eliminar un atributo que se ha añadido a la acción, seleccione Eliminar junto al atributo.
16. Seleccione Agregar acción.
17. Una vez que se hayan agregado todos los tipos de entidad y acciones al esquema, elija Guardar cambios.

JSON mode

Al realizar las actualizaciones, observarás que el editor de JSON valida tu código con la sintaxis de JSON e identificará los errores y las advertencias a medida que lo edites, lo que te permitirá encontrar los problemas rápidamente. Además, no tienes que preocuparte por el formato del JSON, simplemente elige Formato JSON una vez que hayas realizado las actualizaciones, el formato se actualizará para que coincida con el formato JSON esperado.

Para editar un esquema de Verified Permissions en modo JSON

1. Abre el Consola de permisos verificados. Elige tu tienda de pólizas.
2. En el panel de navegación de la izquierda, seleccione Esquema.
3. Seleccione Modo JSON y, a continuación, elija Editar esquema.
4. Introduzca el contenido de su esquema JSON en el campo Contenido. No puede guardar las actualizaciones de su esquema hasta que resuelva todos los errores de sintaxis. Puede elegir Formato JSON para dar formato a la sintaxis JSON de su esquema con el espaciado y la sangría recomendados.
5. Seleccione Save changes (Guardar cambios).

Activación del modo de validación de la política de permisos verificados de Amazon

Puede configurar el modo de validación de políticas en Verified Permissions para controlar si los cambios en las políticas se validan con respecto al esquema de su almacén de políticas.

> **Important**
>
> Al activar la validación de políticas, todos los intentos de crear o actualizar una política o una plantilla de política se validan con el esquema del almacén de políticas. Verified Permissions rechaza el intento de solicitud si la validación falla. Por este motivo, te recomendamos dejar la validación desactivada mientras estás desarrollando la aplicación y activarla para probarla y dejarla activada mientras la aplicación esté en producción.

Consola de administración de AWS

Para configurar el modo de validación de políticas de un almacén de políticas

1. Abre la consola de permisos verificados. Elige tu almacén de políticas.
2. Elija Configuración.
3. En la sección Modo de validación de políticas, seleccione Modificar.
4. Realice una de las siguientes acciones:
   - Para activar la validación de políticas y hacer que todos los cambios en las políticas se validen según su esquema, pulse el botón de opción Estricto (recomendado).
   - Para desactivar la validación de políticas en caso de cambios en las políticas, pulse el botón de opción Desactivada. Escriba `confirm` para confirmar que las actualizaciones de las políticas ya no se validarán según su esquema.
5. Elija Guardar cambios.

AWS CLI

Para configurar el modo de validación de un almacén de políticas

Puede cambiar el modo de validación de un almacén de políticas mediante la UpdatePolicyStoreoperación y especificando un valor diferente para el ValidationSettingsparámetro.

```
$ aws verifiedpermissions update-policy-store \
    --validation-settings "mode=OFF",
    --policy-store-id PSEXAMPLEabcdefg111111
{
    "createdDate": "2023-05-17T18:36:10.134448+00:00",
    "lastUpdatedDate": "2023-05-17T18:36:10.134448+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "validationSettings": {
        "Mode": "OFF"
    }
}
```

Para obtener información más detallada consulte Validación de políticas en la Guía de referencia sobre el lenguaje de las políticas de Cedar.

Políticas de Amazon Verified Permissions

Una política es una instrucción que permite o prohíbe a una entidad principal realizar una o más acciones en un recurso. Cada política se evalúa de forma independiente de las demás políticas. Para obtener más información sobre cómo se estructuran y evalúan las políticas de Cedar, consulte la sección sobre la validación de las políticas de Cedar con el esquema en la Guía de referencia sobre el lenguaje de las políticas de Cedar.

Si lo desea, puede asignar un nombre de política a una política. Los nombres de las políticas deben ser exclusivos para todas las políticas del almacén de políticas y deben ir precedidos de `name/`. Puede usar un nombre de política en lugar del ID de la política en las operaciones del plano de control que acepten un `policyId` parámetro. El siguiente ejemplo usa un nombre de política para recuperar una política con `GetPolicy`.

```
$ aws verifiedpermissions get-policy \
    --policy-id name/example-policy \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Important

Cuando redacte las políticas de Cedar que hagan referencia a las entidades principales, los recursos y las acciones, puede definir los identificadores únicos que se utilizan para cada uno de esos elementos. Le recomendamos encarecidamente que siga las siguientes prácticas recomendadas:

- Utilice identificadores únicos universales (UUID) para todos los identificadores principales y de recursos.

  Por ejemplo, si el usuario `jane` deja la empresa y luego permite que otra persona use el nombre `jane`, ese nuevo usuario tendrá acceso automáticamente a todo lo que otorgan las políticas que aún hacen referencia a `User::"jane"`. Cedar no puede distinguir entre el usuario nuevo y el antiguo. Esto también se aplica a los identificadores de las entidades principales y de los recursos. Utilice siempre identificadores con garantías de que son únicos y que no se han reutilizado nunca para asegurarse de no conceder acceso involuntariamente debido a la presencia de un identificador antiguo en una política.

  Cuando utilice un UUID para una entidad, le recomendamos que lo siga del especificador// comentario y del nombre “descriptivo” de la entidad. Esto ayuda a que sus políticas sean

más fáciles de entender. Por ejemplo: principal == Role: :"A1B2C3D4-E5F6-A1B2-C3D4-example11111",//administradores

- No incluya información de identificación personal, confidencial o sensible como parte del identificador único de sus entidades principales o recursos. Estos identificadores se incluyen en las entradas de registro compartidas en AWS CloudTrail senderos.

Temas

- Creación de políticas estáticas de Amazon Verified Permissions
- Edición de políticas estáticas de Amazon Verified Permissions
- Agregar contexto
- Uso del banco de pruebas de permisos verificados de Amazon
- Tamaño de la política por recurso
- Ejemplo de políticas de Amazon Verified Permissions

Creación de políticas estáticas de Amazon Verified Permissions

Puedes crear una política estática para que los directores les permitan o prohíban realizar acciones específicas en los recursos específicos de tu aplicación. Una política estática incluye valores específicos para `principal` y `resource` y están listos para usarse en las decisiones de autorización.

Consola de administración de AWS

Para crear una política estática

1. Abra el Consola de permisos verificados. Elige tu tienda de pólizas.
2. En el panel de navegación de la izquierda, elija Políticas.
3. Seleccione Crear política y, a continuación, elija Crear política estática.

> **Note**
>
> Si tienes una declaración de política que te gustaría usar, dirígete a Paso 8 y pegue la política en Política sección en la página siguiente.

4. En la sección Efecto de la política, seleccione si la política permitirá o prohibirá una acción cuando una solicitud coincida con la política. Si eliges Permiso, la política permite a los directores realizar las acciones sobre los recursos. Por el contrario, si elige Prohibir, la política no permite a los directores realizar las acciones en los recursos.
5. En el campo Ámbito de las entidades principales, elija el ámbito de las entidades principales al que se aplicará la política.
   - Elija Entidad principal específica para aplicar la política a una entidad principal concreta. Especifique el tipo de entidad y el identificador del principal al que se le permitirá o prohibirá realizar las acciones especificadas en la política.
   - Seleccione Grupo de entidades principales para aplicar la política a un grupo de entidades principales. Escriba el nombre del grupo de entidades principales en el campo Grupo de entidades principales.
   - Seleccione Todas las entidades principales para aplicar la política a todas las entidades principales de su almacén de políticas.
6. En el campo Ámbito de los recursos, elija el ámbito de los recursos al que se aplicará la política.
   - Seleccione Recursos específicos para aplicar la política a un recurso específico. Especifique el tipo de entidad y el identificador del recurso al que se debe aplicar la política.
   - Seleccione Grupo de recursos para aplicar la política a un grupo de recursos. Escriba el nombre del grupo de recursos en el campo Grupo de recursos.
   - Seleccione Todos los recursos para aplicar la política a todos los recursos de su almacén de políticas.
7. En el campo Ámbito de las acciones, elija el ámbito de los recursos al que se aplicará la política.
   - Seleccione Conjunto específico de acciones para aplicar la política a un conjunto de acciones. Seleccione las casillas de verificación situadas junto a las acciones para aplicar la política.
   - Seleccione Todas las acciones para aplicar la política a todas las acciones de su almacén de políticas.
8. Elija Siguiente.

9. En la sección Política, revise su política de Cedar. Puede elegir Formato para dar formato a la sintaxis de su política con el espaciado y la sangría recomendados. Para obtener información más detallada consulte el tema sobre la construcción básica de políticas en Cedar en la Guía de referencia sobre el lenguaje de las políticas de Cedar.
10. En la sección Detalles, escriba una descripción opcional de la política.
11. Elija Crear política.

AWS CLI

Para crear una política estática

Puede crear una política estática mediante el CreatePolicy operación. En el ejemplo siguiente se crea una política estática sencilla.

```
$ aws verifiedpermissions create-policy \
    --definition "{ \"static\": { \"Description\": \"MyTestPolicy\", \"Statement\":
 \"permit(principal,action,resource) when {principal.owner == resource.owner};\"}}"
 \
    --policy-store-id PSEXAMPLEabcdefg111111
{
"Arn": "arn:aws:verifiedpermissions::123456789012:policy/PSEXAMPLEabcdefg111111/
SPEXAMPLEabcdefg111111",
    "createdDate": "2023-05-16T20:33:01.730817+00:00",
    "lastUpdatedDate": "2023-05-16T20:33:01.730817+00:00",
    "policyId": "SPEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "STATIC"
}
```

Para crear una política con un nombre de política

Si lo desea, puede especificar un nombre de política al crear una política. El nombre debe ser único para todas las políticas del almacén de políticas y debe tener el prefijo `name/`. Puedes usar el nombre en lugar del ID de la política.

```
$ aws verifiedpermissions create-policy \
    --definition "{ \"static\": { \"Statement\": \"permit(principal, action,
 resource in Album::\\\"public_folder\\\");\"}}" \
    --policy-store-id PSEXAMPLEabcdefg111111 \
    --name name/example-policy
```

```
{
    "createdDate": "2023-06-12T20:33:37.382907+00:00",
    "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
    "policyId": "SPEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "STATIC",
    "resource": {
        "entityId": "public_folder",
        "entityType": "Album"
    }
}
```

> **Note**
>
> Si especificas un nombre que ya está asociado a otra política en el almacén de políticas, recibirás un `ConflictException` error.

Edición de políticas estáticas de Amazon Verified Permissions

Puedes editar una política estática existente en tu almacén de políticas. Solo puedes actualizar directamente las políticas estáticas. Para cambiar una política vinculada a una plantilla, debe actualizar la plantilla de política. Para obtener más información, consulte Edición de plantillas de política de permisos verificados de Amazon.

Puede cambiar los siguientes elementos de una política estática:

- La `action` referenciada por la política.
- Una cláusula de condición, como `when` y `unless`.

No puedes cambiar los siguientes elementos de una política estática. Para cambiar cualquiera de estos elementos, tendrás que eliminar y volver a crear la política.

- Una política que pasa de ser una política estática a una política vinculada a una plantilla.
- El efecto de una política estática desde `permit` o `forbid`.
- La `principal` a la que hace referencia una política estática.
- El `resource` al que hace referencia una política estática.

Consola de administración de AWS

Para editar una política estática

1. Abre el Consola de permisos verificados. Elige tu tienda de pólizas.
2. En el panel de navegación de la izquierda, seleccione Políticas.
3. Seleccione el botón de opción situado junto a la política estática que desee editar y, a continuación, Editar.
4. En la sección Cuerpo de la política, actualice `action` o la cláusula de condición de su política estática. No puede actualizar el efecto de la política, `principal`, o `resource` de la política.
5. Elija Actualizar política.

> **Note**
>
> Si la validación de políticas está habilitada en el almacén de políticas, la actualización de una política estática hace que Verified Permissions valide la política con el esquema del almacén de políticas. Si la política estática actualizada no supera la validación, se produce un error en la operación y la actualización no se guarda.

AWS CLI

Para editar una política estática

Puede editar una política estática mediante el UpdatePolicy operación. En el ejemplo siguiente se edita una política estática sencilla.

En el ejemplo, se utiliza el archivo `definition.txt` para incluir la definición de la política.

```
{
    "static": {
        "description":  "Grant everyone of janeFriends UserGroup access to the
 vacationFolder Album",
        "statement": "permit(principal in UserGroup::\"janeFriends\", action,
 resource in Album::\"vacationFolder\" );"
    }
}
```

El siguiente comando hace referencia a ese archivo.

```
$ aws verifiedpermissions create-policy \
    --definition file://definition.txt \
    --policy-store-id PSEXAMPLEabcdefg111111

{
    "createdDate": "2023-06-12T20:33:37.382907+00:00",
    "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
    "policyId": "SPEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "STATIC",
    "principal": {
        "entityId": "janeFriends",
        "entityType": "UserGroup"
    },
    "resource": {
        "entityId": "vacationFolder",
        "entityType": "Album"
    }
}
```

Para actualizar el nombre de una política

Puede establecer o actualizar el nombre de una política al actualizar una política. El nombre debe ser único para todas las políticas del almacén de políticas y debe tener el prefijo `name/`. Si no incluyes el campo de nombre en la solicitud de actualización, el nombre existente no cambiará. Para eliminar un nombre, configúralo en una cadena vacía.

```
$ aws verifiedpermissions update-policy \
    --policy-id SPEXAMPLEabcdefg111111 \
    --policy-store-id PSEXAMPLEabcdefg111111 \
    --definition file://definition.txt \
    --name name/example-policy
{
    "createdDate": "2023-06-12T20:33:37.382907+00:00",
    "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
    "policyId": "SPEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "STATIC",
    "principal": {
        "entityId": "janeFriends",
```

```
        "entityType": "UserGroup"
    },
    "resource": {
        "entityId": "vacationFolder",
        "entityType": "Album"
    }
}
```

Agregar contexto

Contexto es la información que es relevante para las decisiones políticas, pero no forma parte de la identidad de su director, acción o recurso. Las reclamaciones del token de acceso son de contexto. Es posible que desee permitir una acción solo desde un conjunto de direcciones IP de origen o solo si el usuario ha iniciado sesión con MFA. La aplicación tiene acceso a estos datos contextuales de sesión y debe rellenarlos para las solicitudes de autorización. Los datos de contexto de una solicitud de autorización de permisos verificados deben ser JSON-formatted en un `contextMap` elemento.

Los ejemplos que ilustran este contenido provienen de tienda de políticas de muestra. Para seguir adelante, crea el DigitalPetStore tienda de políticas de muestra en su entorno de pruebas.

El siguiente objeto de contexto declara uno de cada tipo de datos de Cedar para una aplicación basándose en el ejemplo DigitalPetStore tienda de políticas.

```
"context": {
  "contextMap": {
    "AccountCodes": {
      "set": [
        {
          "long": 111122223333
        },
        {
          "long": 444455556666
        },
        {
          "long": 123456789012
        }
      ]
    },
    "approvedBy": {
    "entityIdentifier": {
      "entityId": "Bob",
```

```
      "entityType": "DigitalPetStore::User"
    }
    },
    "MfaAuthorized": {
      "boolean": true
    },
    "NetworkInfo": {
      "record": {
        "IPAddress": {
          "string": "192.0.2.178"
        },
        "Country": {
          "string": "United States of America"
        },
        "SSL": {
          "boolean": true
        }
    }
    },
    "RequestedOrderCount": {
      "long": 4
    },
    "UserAgent": {
      "string": "My UserAgent 1.12"
    }
  }
}
```

Tipos de datos en el contexto de la autorización

Booleano

Un binario `true` o `false` valor. En el ejemplo, el valor booleano de `true` por `MfaAuthenticated` indica que el cliente ha realizado una autenticación multifactorial antes de solicitar la visualización de su pedido.

Establezca

Una colección de elementos de contexto. Los miembros del conjunto pueden ser todos del mismo tipo, como en este ejemplo, o de tipos diferentes, incluido un conjunto anidado. En el ejemplo, el cliente está asociado a 3 cuentas diferentes.

Cadena

Una secuencia de letras, números o símbolos, encerrados en '' caracteres. En el ejemplo, el `UserAgent` la cadena representa el navegador que el cliente utilizó para solicitar la visualización de su pedido.

Largo

Un número entero. En el ejemplo, el `RequestedOrderCount` indica que esta solicitud forma parte de un lote que se produjo cuando el cliente solicitó ver cuatro de sus pedidos anteriores.

Registro

Una colección de atributos. Debe declarar estos atributos en el contexto de la solicitud. Un almacén de políticas con un esquema debe incluir esta entidad y los atributos de la entidad en el esquema. En el ejemplo, el `NetworkInfo` el registro contiene información sobre la IP de origen del usuario, la geolocalización de esa IP según lo determine el cliente y el cifrado en tránsito.

EntityIdentifier

Una referencia a una entidad y a los atributos declarados en el `entities` elemento de la solicitud. En el ejemplo, el pedido del usuario fue aprobado por el empleado `Bob`.

Para probar este contexto de ejemplo en el ejemplo DigitalPetStore aplicación, debes actualizar tu solicitud `entities`, el esquema del almacén de políticas y la política estática con la descripción Función de cliente: obtener un pedido.

Modificando DigitalPetStore para aceptar el contexto de autorización

Inicialmente, DigitalPetStore no es un almacén de políticas muy complejo. No incluye políticas ni atributos de contexto preconfigurados para respaldar el contexto que hemos presentado. Para evaluar un ejemplo de solicitud de autorización con esta información de contexto, realiza las siguientes modificaciones en tu almacén de políticas y en tu solicitud de autorización. Para ver ejemplos de contexto con la información del token de acceso como contexto, consulte Cartografía Amazon Cognito tokens de acceso y Mapeo de los tokens de acceso de OIDC.

Schema

Aplica las siguientes actualizaciones al esquema de tu almacén de políticas para admitir los nuevos atributos de contexto. Actualice `GetOrder` en `actions` de la siguiente manera.

```
"GetOrder": {
```

```
  "memberOf": [],
  "appliesTo": {
    "resourceTypes": [
      "Order"
    ],
    "context": {
      "type": "Record",
      "attributes": {
        "AccountCodes": {
          "type": "Set",
          "required": true,
          "element": {
            "type": "Long"
          }
        },
        "approvedBy": {
          "name": "User",
          "required": true,
          "type": "Entity"
        },
        "MfaAuthorized": {
          "type": "Boolean",
          "required": true
        },
        "NetworkInfo": {
          "type": "NetworkInfo",
          "required": true
        },
        "RequestedOrderCount": {
          "type": "Long",
          "required": true
        },
        "UserAgent": {
          "required": true,
          "type": "String"
        }
      }
    },
    "principalTypes": [
      "User"
    ]
  }
}
```

Para hacer referencia al `record` tipo de datos denominado `NetworkInfo` en el contexto de su solicitud, cree un CommonType construya en su esquema agregando lo siguiente a su esquema antes `actions`. UN `commonType` construir es un conjunto compartido de atributos que se pueden aplicar a diferentes entidades.

```
"commonTypes": {
  "NetworkInfo": {
    "attributes": {
      "IPAddress": {
        "type": "String",
        "required": true
      },
      "SSL": {
        "required": true,
        "type": "Boolean"
      },
      "Country": {
        "required": true,
        "type": "String"
      }
    },
    "type": "Record"
  }
},
```

Policy

La siguiente política establece las condiciones que deben cumplir cada uno de los elementos de contexto proporcionados. Se basa en la política estática existente con la descripción Función del cliente: obtener un pedido. Inicialmente, esta política solo requiere que el principal que hace una solicitud sea el propietario del recurso.

```
permit (
    principal in DigitalPetStore::Role::"Customer",
    action in [DigitalPetStore::Action::"GetOrder"],
    resource
) when {
    principal == resource.owner &&
    context.AccountCodes.contains(111122223333) &&
    context.approvedBy in DigitalPetStore::Role::"Employee" &&
    context.MfaAuthorized == true &&
    context.NetworkInfo.Country like "*United States*" &&
```

```
    context.NetworkInfo.IPAddress like "192.0.2.*" &&
    context.NetworkInfo.SSL == true &&
    context.RequestedOrderCount <= 4 &&
    context.UserAgent like "*My UserAgent*"
};
```

Ahora exigimos que la solicitud para recuperar un pedido cumpla con las condiciones de contexto adicionales que agregamos a la solicitud.

1. El usuario debe haber iniciado sesión con MFA.
2. El navegador web del usuario `User-Agent` debe contener la cadena `My UserAgent`.
3. El usuario debe haber solicitado ver 4 pedidos o menos.
4. Uno de los códigos de cuenta del usuario debe ser `111122223333`.
5. La dirección IP del usuario debe originarse en los Estados Unidos, debe estar en una sesión cifrada y su dirección IP debe empezar por `192.0.2.`.
6. Un empleado debe haber aprobado su pedido. En el `entities` elemento de la solicitud de autorización, declararemos un usuario `Bob` quien tiene el rol de `Employee`.

Request body

Tras configurar el almacén de políticas con el esquema y la política adecuados, puedes presentar esta solicitud de autorización a la operación de la API de permisos verificados IsAuthorized. Tenga en cuenta que el `entities` el segmento contiene una definición de `Bob`, un usuario con el rol de `Employee`.

```
{
  "principal": {
    "entityType": "DigitalPetStore::User",
    "entityId": "Alice"
  },
  "action": {
    "actionType": "DigitalPetStore::Action",
    "actionId": "GetOrder"
  },
  "resource": {
    "entityType": "DigitalPetStore::Order",
    "entityId": "1234"
  },
  "context": {
```

```
    "contextMap": {
      "AccountCodes": {
        "set": [
          {"long": 111122223333},
          {"long": 444455556666},
          {"long": 123456789012}
        ]
      },
      "approvedBy": {
        "entityIdentifier": {
          "entityId": "Bob",
          "entityType": "DigitalPetStore::User"
        }
      },
      "MfaAuthorized": {
        "boolean": true
      },
      "NetworkInfo": {
        "record": {
          "Country": {"string": "United States of America"},
          "IPAddress": {"string": "192.0.2.178"},
          "SSL": {"boolean": true}
        }
      },
      "RequestedOrderCount":{
        "long": 4
      },
      "UserAgent": {
        "string": "My UserAgent 1.12"
      }
    }
  },
  "entities": {
    "entityList": [
      {
        "identifier": {
          "entityType": "DigitalPetStore::User",
          "entityId": "Alice"
        },
        "attributes": {
          "memberId": {
            "string": "801b87f2-1a5c-40b3-b580-eacad506d4e6"
          }
        },
```

```
        "parents": [
          {
            "entityType": "DigitalPetStore::Role",
            "entityId": "Customer"
          }
        ]
      },
      {
        "identifier": {
          "entityType": "DigitalPetStore::User",
          "entityId": "Bob"
        },
        "attributes": {
          "memberId": {
            "string": "49d9b81e-735d-429c-989d-93bec0bcfd8b"
          }
        },
        "parents": [
          {
            "entityType": "DigitalPetStore::Role",
            "entityId": "Employee"
          }
        ]
      },
      {
        "identifier": {
          "entityType": "DigitalPetStore::Order",
          "entityId": "1234"
        },
        "attributes": {
          "owner": {
            "entityIdentifier": {
              "entityType": "DigitalPetStore::User",
              "entityId": "Alice"
            }
          }
        },
        "parents": []
      }
     ]
   },
   "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

Uso del banco de pruebas de permisos verificados de Amazon

Utilice el banco de pruebas de permisos verificados para probar y solucionar problemas con las políticas de permisos verificados ejecutando solicitudes de autorización contra ellos. El banco de pruebas utiliza los parámetros que usted especifique para determinar si las políticas de Cedar de su almacén de políticas autorizarían la solicitud. Puede cambiar entre el modo visual y el modo JSON mientras prueba las solicitudes de autorización. Para obtener más información sobre cómo se estructuran y evalúan las políticas de Cedar, consulte la sección sobre la construcción básica de políticas en Cedar en la Guía de referencia sobre el lenguaje de las políticas de Cedar.

> **Note**
>
> Al realizar una solicitud de autorización mediante Verified Permissions, puede proporcionar la lista de entidades principales y recursos como parte de la solicitud en la sección Entidades adicionales. Sin embargo, no es posible incluir los detalles de las acciones. Deben especificarse en el esquema o deducirse de la solicitud. No puede incluir una acción en la sección Entidades adicionales.

Para obtener una descripción visual y una demostración del banco de pruebas, consulte Permisos verificados de Amazon: creación y prueba de políticas (Primer Series #3) en el AWS YouTube canal.

Visual mode

> **Note**
>
> Debe tener un esquema definido en su almacén de políticas para utilizar el modo visual del banco de pruebas.

Para probar las políticas en modo visual

1. Abre el Consola de permisos verificados. Elige tu tienda de pólizas.
2. En el panel de navegación de la izquierda, seleccione Banco de pruebas.
3. Elija Modo visual.
4. En la sección Entidad principal, elija Entidad principal que realiza la acción entre los tipos de entidad principal del esquema. Escriba un identificador para la entidad principal en el cuadro de texto.

5. (Opcional) Seleccione Agregar un elemento principal para añadir las entidades principales del elemento principal especificado. Para eliminar un elemento principal que se haya agregado a la entidad principal, seleccione Eliminar junto al nombre del elemento principal.
6. Especifique el valor de atributo para cada atributo de la entidad principal especificada. El banco de pruebas utiliza los valores de atributo especificados en la solicitud de autorización simulada.
7. En la sección Recursos, elija el recurso sobre el que actúa la entidad principal. Escriba un identificador para el recurso en el cuadro de texto.
8. (Opcional) Seleccione Agregar un elemento principal para añadir las entidades principales del recurso especificado. Para eliminar un elemento principal que se haya agregado al recurso, seleccione Eliminar junto al nombre del elemento principal.
9. Especifique el valor de atributo para cada atributo del recurso especificado. El banco de pruebas utiliza los valores de atributo especificados en la solicitud de autorización simulada.
10. En la sección Acción, elija Acción que va a tomar la entidad principal en la lista de acciones válidas para la entidad principal y el recurso especificados.
11. Especifique el valor de atributo para cada atributo de la acción especificada. El banco de pruebas utiliza los valores de atributo especificados en la solicitud de autorización simulada.
12. (Opcional) En la sección Entidades adicionales, elija Agregar entidad para agregar las entidades que se evaluarán para la decisión de autorización.
13. Elija el identificador de entidad en la lista desplegable y escriba el identificador de entidad.
14. (Opcional) Seleccione Agregar un elemento principal para añadir las entidades principales de la entidad especificada. Para eliminar un elemento principal que se haya agregado a la entidad, seleccione Eliminar junto al nombre del elemento principal.
15. Especifique el valor de atributo para cada atributo de la entidad especificada. El banco de pruebas utiliza los valores de atributo especificados en la solicitud de autorización simulada.
16. Elija Confirmar para añadir la entidad al banco de pruebas.
17. Seleccione Ejecutar solicitud de autorización para simular la solicitud de autorización para las políticas de Cedar en el almacén de políticas. El banco de pruebas muestra la decisión de aceptar o denegar la solicitud junto con información sobre las políticas satisfechas o los errores encontrados durante la evaluación.

JSON mode

Para probar las políticas en modo JSON

1. Abre el Consola de permisos verificados. Elige tu tienda de pólizas.
2. En el panel de navegación de la izquierda, seleccione Banco de pruebas.
3. Elija Modo JSON.
4. En la sección Detalles de la solicitud, si tiene un esquema definido, elija Entidad principal que realiza la acción entre los tipos de entidad principal de su esquema. Escriba un identificador para la entidad principal en el cuadro de texto.

   Si no tiene un esquema definido, escriba la entidad principal en el cuadro de texto Entidad principal que realiza la acción.
5. Si tiene un esquema definido, elija el recurso entre los tipos de recurso del esquema. Escriba un identificador para el recurso en el cuadro de texto.

   Si no tiene un esquema definido, escriba el recurso en el cuadro de texto Recurso.
6. Si tiene un esquema definido, elija la acción en la lista de acciones válidas para la entidad principal y el recurso especificados.

   Si no tiene un esquema definido, escriba el recurso en el cuadro de texto Acción.
7. Introduzca el contexto de la solicitud de simulación en el campo Contexto. El contexto de la solicitud es información adicional que se puede utilizar para tomar decisiones de autorización.
8. En el campo Entidades, introduzca la jerarquía de las entidades y los atributos que se van a evaluar para la decisión de autorización.
9. Seleccione Ejecutar solicitud de autorización para simular la solicitud de autorización para las políticas de Cedar en el almacén de políticas. El banco de pruebas muestra la decisión de aceptar o denegar la solicitud junto con información sobre las políticas satisfechas o los errores encontrados durante la evaluación.

Tamaño de la política por recurso

Los permisos verificados limitan el tamaño total de todas las políticas que hacen referencia al mismo recurso. Este límite es el Tamaño de la política por recurso cuota y su valor predeterminado es de 1 MB. Cuando llamas `CreatePolicy` o `UpdatePolicy`, Verified Permissions calcula el nuevo total

del recurso al que se hace referencia. Si el nuevo total supera la cuota, se produce un error en la solicitud `ServiceQuotaExceededException`.

Los permisos verificados también limitan el tamaño total de todas las políticas que hacen referencia a la misma combinación de un principal y un recurso. Este límite es el Combinación del tamaño de la política por recurso principal cuota y su valor es de 200 000 bytes. Este límite no se puede ajustar.

Las políticas que no especifican un recurso se aplican a todos los recursos. Estas políticas comparten un total independiente para el `"unspecified"` recurso, que está sujeto a lo mismo Tamaño de la política por recurso cuota.

Para ver la lista de todas las cuotas de permisos verificados, consulta Cuotas para Amazon Verified Permissions.

Temas

- Cómo se calculan los tamaños de las pólizas
- Mantenerse dentro de la cuota
- Template-linked ejemplo de tamaño de política

Cómo se calculan los tamaños de las pólizas

Cada política contribuye al total de un recurso de la siguiente manera:

- Políticas estáticas — Una política estática aporta su tamaño total. Se trata del mismo valor que se compara con el Tamaño de la política cuota.
- Template-linked políticas — Una política vinculada a una plantilla aporta la longitud combinada del principal y el recurso utilizado para instanciarla, así como algunos gastos fijos almacenados en el elemento de la base de datos. Si una plantilla no tiene un espacio principal o de recursos, la longitud de esta entidad no especificada se contabiliza como 0. El tamaño del cuerpo de la plantilla de políticas en sí no se tiene en cuenta para esta cuota.

Cada política también añade una sobrecarga fija de hasta unos pocos cientos de bytes al total. Template-linked las políticas añaden más gastos generales que las políticas estáticas.

Mantenerse dentro de la cuota

Si su solicitud se acerca al Tamaño de la política por recurso cuota, considere las siguientes estrategias:

- Utilice el tipo de política apropiado. Si muchas políticas conceden permisos similares sobre el mismo recurso, las plantillas de políticas pueden reducir el consumo de almacenamiento del almacén de políticas. Cada política vinculada a una plantilla almacena solo su contenido principal y sus recursos. La plantilla de política almacena la declaración compartida una vez, pero cada política vinculada también conlleva una sobrecarga de almacenamiento fija. Las plantillas solo se amortizan cuando la declaración compartida es lo suficientemente grande como para compensar esa sobrecarga, aproximadamente 200 bytes o más. Por debajo de ese umbral, las políticas estáticas individuales consumen menos espacio. Para obtener más información, consulte Plantillas de políticas de permisos verificados de Amazon y políticas vinculadas a plantillas.
- Especifique los principales y los recursos en el ámbito de la política. Las políticas que no especifican un recurso comparten un único total para `"unspecified"` recurso en todo el almacén de políticas. Al asignar el alcance de cada política a un recurso específico, las políticas se distribuyen en totales distintos, uno para cada recurso.
- Agrupe sus principales y recursos. En lugar de escribir una política para cada director, escribe una política para un grupo y agrega directores al grupo. Agrupar los recursos en contenedores también puede reducir la cantidad de políticas que hacen referencia a un solo recurso.
- Solicitar un aumento de cuota. Puede solicitar un aumento del Tamaño de la póliza por recurso cuota si el diseño de su política cumple con ciertas restricciones. Para cumplir los requisitos, las políticas deben especificar el principal o el recurso incluido en el ámbito de la política. Un aumento también podría reducir la cantidad de padres jerárquicos que pueden tener sus directores y recursos. El Tamaño de la política por combinación principal de recursos la cuota es un límite independiente de 200 000 bytes que se aplica a las políticas que hacen referencia a la misma combinación de principal y recurso. Este límite no se puede ajustar.

Template-linked ejemplo de tamaño de política

Para saber cómo las políticas vinculadas a plantillas contribuyen a la Tamaño de la política por recurso la cuota y el Tamaño de la política por combinación principal de recursos cuota, suma la longitud del principal y la longitud del recurso. Si no especificas un recurso, Verified Permissions contabiliza su tamaño para el `"unspecified"` recurso.

Revisa la siguiente plantilla:

```
permit (
  principal in ?principal,
  action in
    [MyApplication::Action::"ContentManagement",
```

```
      MyApplication::Action::"AccountAdministration",
      MyApplication::Action::"BillingOperations"],
  resource in ?resource
)
when {
  resource has authorizedTenantIds &&
  principal has tenantId &&
  resource.authorizedTenantIds.contains(principal.tenantId)
};
```

Cree las siguientes políticas a partir de esa plantilla:

```
TemplateLinkedPolicy {
  policyId: "policy1",
  templateId: "template1",
  principal: User::"alice",
  resource: Photo::"car.jpg"
}

TemplateLinkedPolicy {
  policyId: "policy2",
  templateId: "template1",
  principal: User::"bob",
  resource: Photo::"boat.jpg"
}

TemplateLinkedPolicy {
  policyId: "policy3",
  templateId: "template1",
  principal: User::"jane",
  resource: Photo::"car.jpg"
}

TemplateLinkedPolicy {
  policyId: "policy4",
  templateId: "template1",
  principal: User::"jane",
  resource: Photo::"boat.jpg"
}

TemplateLinkedPolicy {
  policyId: "policy5",
  templateId: "template1",
```

```
  principal: User::"jane",
  resource
}
```

Calcule el tamaño de esas políticas contando los bytes del `principal` y `resource` para cada uno. Los permisos verificados miden el tamaño en UTF-8 bytes. Los caracteres del rango ASCII imprimible cuentan como 1 byte cada uno, pero otros caracteres pueden usar más. En los ejemplos siguientes solo se utilizan caracteres ASCII, por lo que cada carácter cuenta como 1 byte. Estos cálculos excluyen la sobrecarga fija. Sus totales reales serán más altos.

El tamaño de `policy1` es la longitud del principal `User::"alice"` (13) más la longitud del recurso `Photo::"car.jpg"` (16). El total es 13 + 16 = 29 bytes.

El tamaño de `policy2` es la longitud del principal `User::"bob"` (11) más la longitud del recurso `Photo::"boat.jpg"` (17). El total es 11 + 17 = 28 bytes.

El tamaño de `policy3` es la longitud del principal `User::"jane"` (12) más la longitud del recurso `Photo::"car.jpg"` (16). El total es 12 + 16 = 28 bytes.

El tamaño de `policy4` es la longitud del principal `User::"jane"` (12) más la longitud del recurso `Photo::"boat.jpg"` (17). El total es 12 + 17 = 29 bytes.

El tamaño de `policy5` es la longitud del principal `User::"jane"` (12) más la longitud del recurso (0). El total es 12 + 0 = 12 bytes.

Los siguientes totales se aplican a Tamaño de la política por recurso cuota. Porque `policy5` no especifica un recurso, su tamaño cuenta para el `"unspecified"` recurso.

- El total del `"unspecified"` el recurso es `policy5` (12), que es de 12 bytes.
- El total de `Photo::"car.jpg"` es la suma de `policy1` (29) y `policy3` (28), que es de 57 bytes.
- El total de `Photo::"boat.jpg"` es la suma de `policy2` (28) y `policy4` (29), que es de 57 bytes.

Los siguientes totales se aplican a Combinación del tamaño de la póliza por recurso principal cuota. Porque `policy5` no especifica un recurso, su tamaño cuenta para la combinación de su principal, `User::"jane"`, y el `"unspecified"` recurso.

- El total de `User::"alice"` y `Photo::"car.jpg"` es `policy1` (29), que es de 29 bytes.

- El total de `User::"bob"` y `Photo::"boat.jpg"` es `policy2` (28), que es de 28 bytes.
- El total de `User::"jane"` y `Photo::"car.jpg"` es `policy3` (28), que es de 28 bytes.
- El total de `User::"jane"` y `Photo::"boat.jpg"` es `policy4` (29), que es de 29 bytes.
- El total de `User::"jane"` y el `"unspecified"` el recurso es `policy5` (12), que es de 12 bytes.

Ejemplo de políticas de Amazon Verified Permissions

Algunos de los ejemplos de políticas que se incluyen aquí son ejemplos básicos de políticas de Cedar y otros están verificados Permissions-specific. Los básicos enlazan a la Guía de referencia del lenguaje de la política de Cedar y se incluyen allí. Para obtener más información sobre la sintaxis de las políticas de Cedar, consulte el tema sobre la construcción básica de políticas en Cedar en la Guía de referencia sobre el lenguaje de las políticas de Cedar.

Ejemplos de políticas

- Permite el acceso a entidades individuales
- Permite el acceso a grupos de entidades
- Permite el acceso a cualquier entidad
- Permite el acceso a los atributos de una entidad (ABAC)
- Denega el acceso
- Usa la notación entre corchetes para hacer referencia a los atributos del token
- Usa la notación de puntos para hacer referencia a los atributos
- Refleja Amazon Cognito Atributos del token de identificación
- Refleja los atributos del token de ID de OIDC
- Refleja Amazon Cognito atributos del token de acceso
- Refleja los atributos del token de acceso de OIDC

Usa la notación entre corchetes para hacer referencia a los atributos del token

El siguiente ejemplo muestra cómo puede crear una política que utilice la notación entre corchetes para hacer referencia a los atributos del token.

Para obtener más información sobre el uso de los atributos de los tokens en las políticas de los permisos verificados, consulta Cartografía Amazon Cognito tokens al esquema y Asignar los tokens OIDC al esquema.

```
permit (
    principal in MyCorp::UserGroup::"us-west-2_EXAMPLE|MyUserGroup",
    action,
    resource
) when {
    principal["cognito:username"] == "alice" &&
    principal["custom:employmentStoreCode"] == "petstore-dallas" &&
    principal has email && principal.email == "alice@example.com" &&
    context["ip-address"] like "192.0.2.*"
};
```

Usa la notación de puntos para hacer referencia a los atributos

El siguiente ejemplo muestra cómo puede crear una política que utilice la notación de puntos para hacer referencia a los atributos.

Para obtener más información sobre el uso de los atributos de token en las políticas de los permisos verificados, consulta Cartografía Amazon Cognito tokens al esquema y Asignar los tokens OIDC al esquema.

```
permit(principal, action, resource)
when {
    principal.cognito.username == "alice" &&
    principal.custom.employmentStoreCode == "petstore-dallas" &&
    principal.tenant == "x11app-tenant-1" &&
    principal has email && principal.email == "alice@example.com"
};
```

Refleja Amazon Cognito Atributos del token de identificación

El siguiente ejemplo muestra cómo puede crear una política que haga referencia a los atributos del token de ID desde Amazon Cognito.

Para obtener más información sobre el uso de atributos de token en las políticas de los permisos verificados, consulta Cartografía Amazon Cognito tokens al esquema y Asignar los tokens OIDC al esquema.

```
permit (
    principal in MyCorp::UserGroup::"us-west-2_EXAMPLE|MyUserGroup",
    action,
    resource
) when {
    principal["cognito:username"] == "alice" &&
    principal["custom:employmentStoreCode"] == "petstore-dallas" &&
    principal.tenant == "x11app-tenant-1" &&
    principal has email && principal.email == "alice@example.com"
};
```

Refleja los atributos del token de ID de OIDC

El siguiente ejemplo muestra cómo se puede crear una política que haga referencia a los atributos del token de ID de un proveedor de OIDC.

Para obtener más información sobre el uso de los atributos de token en las políticas de los permisos verificados, consulta Cartografía Amazon Cognito tokens al esquema y Asignar los tokens OIDC al esquema.

```
permit (
    principal in MyCorp::UserGroup::"MyOIDCProvider|MyUserGroup",
    action,
    resource
) when {
    principal.email_verified == true && principal.email == "alice@example.com" &&
    principal.phone_number_verified == true && principal.phone_number like "+1206*"
};
```

Refleja Amazon Cognito atributos del token de acceso

El siguiente ejemplo muestra cómo puede crear una política que haga referencia a los atributos del token de acceso desde Amazon Cognito.

Para obtener más información sobre el uso de atributos de token en las políticas de los permisos verificados, consulta Cartografía Amazon Cognito tokens al esquema y Asignar los tokens OIDC al esquema.

```
permit(principal, action in [MyApplication::Action::"Read",
 MyApplication::Action::"GetStoreInventory"], resource)
```

```
when {
    context.token.client_id == "52n97d5afhfiu1c4di1k5m8f60" &&
    context.token.scope.contains("MyAPI/mydata.write")
};
```

Refleja los atributos del token de acceso de OIDC

El siguiente ejemplo muestra cómo se puede crear una política que haga referencia a los atributos del token de acceso de un proveedor de OIDC.

Para obtener más información sobre el uso de los atributos de los tokens en las políticas de los permisos verificados, consulta Cartografía Amazon Cognito tokens al esquema y Asignar los tokens OIDC al esquema.

```
permit(
    principal,
    action in [MyApplication::Action::"Read",
 MyApplication::Action::"GetStoreInventory"],
    resource
)
when {
    context.token.client_id == "52n97d5afhfiu1c4di1k5m8f60" &&
    context.token.scope.contains("MyAPI-read")
};
```

Plantillas de políticas de permisos verificados de Amazon y políticas vinculadas a plantillas

En los permisos verificados, las plantillas de políticas son políticas con marcadores de posición para los permisos verificados `principal`, `resource` o para ambos. Las plantillas de políticas por sí solas no se pueden utilizar para gestionar las solicitudes de autorización. Para gestionar las solicitudes de autorización, se debe crear una política vinculada a una plantilla basada en una plantilla de política. Las plantillas de políticas permiten definir una política una vez y, después, utilizarla con varios principios y recursos. Las actualizaciones de la plantilla de política se reflejan en todas las políticas que utilizan la plantilla. Para obtener información más detallada consulte Plantillas de política de Cedar en la Guía de referencia sobre el lenguaje de las políticas de Cedar.

Si lo desea, puede asignar un nombre de plantilla de política a una plantilla de política. Los nombres de las plantillas de políticas deben ser únicos en el almacén de políticas y tener el prefijo. `name/` Puede utilizar un nombre de plantilla de política en lugar del ID de la plantilla de política en las operaciones del plano de control que acepten un `policyTemplateId` parámetro. Solo `GetPolicyTemplate` y `ListPolicyTemplates` devuelve el nombre en la salida. En el siguiente ejemplo, se utiliza un nombre de plantilla de política para recuperar una plantilla de política con`GetPolicyTemplate`.

```
$ aws verifiedpermissions get-policy-template \
    --policy-template-id name/example-policy-template \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Por ejemplo, la siguiente plantilla de política proporciona `Read` y `Comment` permisos para el principal y el recurso que utilizan la plantilla de política. `Edit`

```
permit(
  principal == ?principal,
  action in [Action::"Read", Action::"Edit", Action::"Comment"],
  resource == ?resource
);
```

Si tuviera que crear una política con un nombre `Editor` basado en esta plantilla, cuando se designe a un director como editor de un recurso específico, su aplicación crearía una política que otorgue permisos al director para leer, editar y comentar el recurso.

A diferencia de las políticas estáticas, las políticas vinculadas a plantillas son dinámicas. Tomemos el ejemplo anterior: si eliminara la `Comment` acción de la plantilla de política, cualquier política vinculada a esa plantilla o basada en ella se actualizaría en consecuencia y los directores especificados en las políticas ya no podrían hacer comentarios sobre los recursos correspondientes.

Para ver más ejemplos de políticas vinculadas a plantillas, consulte. Ejemplos de políticas vinculadas a plantillas de permisos verificados de Amazon

Creación de plantillas de políticas de permisos verificados de Amazon

Puede crear plantillas de políticas en Verified Permissions utilizando el Consola de administración de AWS AWS CLI, el o el AWS SDKs. Las plantillas de políticas permiten definir una política una vez y, a continuación, utilizarla con varios principios y recursos. Una vez que haya creado una plantilla de política, podrá crear políticas vinculadas a plantillas para utilizarlas con principios y recursos específicos. Para obtener más información, consulte Creación de políticas vinculadas a plantillas de permisos verificados de Amazon.

Consola de administración de AWS

Creación de una plantilla de política

1. Abra la consola de permisos verificados. Elige tu almacén de políticas.
2. En el panel de navegación de la izquierda, elija Plantillas de política.
3. Elija Crear plantilla de política.
4. En la sección Detalles, escriba una descripción de la plantilla de política.
5. En la sección Cuerpo de la plantilla de política, utilice los marcadores de posición ? `principal` y `?resource` para permitir que las políticas creadas a partir de esta plantilla personalicen los permisos que conceden. Puede elegir Formato para dar formato a la sintaxis de su plantilla de política con el espaciado y la sangría recomendados.
6. Elija Crear plantilla de política.

AWS CLI

Para crear una plantilla de política

Puede crear una plantilla de políticas mediante la CreatePolicyTemplateoperación. En el siguiente ejemplo, se crea una plantilla de política con un marcador de posición para la entidad principal.

El archivo `template1.txt` contiene lo siguiente.

```
"VacationAccess"
permit(
    principal in ?principal,
    action == Action::"view",
    resource == Photo::"VacationPhoto94.jpg"
);
```

```
$ aws verifiedpermissions create-policy-template \
    --description "Template for vacation picture access"
    --statement file://template1.txt
    --policy-store-id PSEXAMPLEabcdefg111111
{
    "createdDate": "2023-05-18T21:17:47.284268+00:00",
    "lastUpdatedDate": "2023-05-18T21:17:47.284268+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyTemplateId": "PTEXAMPLEabcdefg111111"
}
```

Para crear una plantilla de política con un nombre de plantilla de política

Si lo desea, puede especificar un nombre de plantilla de política al crear una plantilla de política. El nombre debe ser único para todas las plantillas de políticas del almacén de políticas y debe tener el prefijo. `name/` Puede usar el nombre en lugar del ID de la plantilla de políticas.

```
$ aws verifiedpermissions create-policy-template \
    --description "Template for vacation picture access" \
    --statement file://template1.txt \
    --policy-store-id PSEXAMPLEabcdefg111111 \
    --name name/example-policy-template
{
    "createdDate": "2023-06-12T20:47:42.804511+00:00",
    "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyTemplateId": "PTEXAMPLEabcdefg111111"
}
```

> **Note**
>
> Si especifica un nombre que ya está asociado a otra plantilla de políticas en el almacén de políticas, recibirá un `ConflictException` error.

Creación de políticas vinculadas a plantillas de permisos verificados de Amazon

Puede crear políticas vinculadas a plantillas, o políticas que se basen en una plantilla de política, utilizando, o. Consola de administración de AWS AWS CLI AWS SDKs Las políticas vinculadas a plantillas permanecen vinculadas a sus plantillas de políticas. Si cambia la declaración de política en la plantilla de política, cualquier política vinculada a esa plantilla utilizará automáticamente la nueva declaración para todas las decisiones de autorización que se tomen a partir de ese momento.

Para ver ejemplos de políticas vinculadas a plantillas, consulte. Ejemplos de políticas vinculadas a plantillas de permisos verificados de Amazon

Consola de administración de AWS

Para crear una política vinculada a una plantilla mediante la creación de una instancia de plantilla de política

1. Abra la consola de permisos verificados. Elige tu almacén de políticas.
2. En el panel de navegación de la izquierda, elija Políticas.
3. Seleccione Crear política y, a continuación, elija Crear política vinculada a una plantilla.
4. Seleccione el botón de opción situado junto a la plantilla de política que desee utilizar y, a continuación, elija Siguiente.
5. Escriba la entidad principal y el recurso que desea utilizar para esta instancia específica de la política vinculada a la plantilla. Los valores especificados se muestran en el campo de vista previa de la instrucción de política.

   > **Note**
   >
   > Los valores de Entidad principal y Recurso deben tener el mismo formato que las políticas estáticas. Por ejemplo, para especificar el grupo `AdminUsers` de la entidad

principal, escriba `Group::"AdminUsers"`. Si escribe `AdminUsers`, se muestra un error de validación.

6. Seleccione Crear política vinculada a una plantilla.

   La nueva política vinculada a una plantilla se muestra en Políticas.

AWS CLI

Para crear una política vinculada a una plantilla mediante la creación de una instancia de plantilla de política

Puede crear una política vinculada a una plantilla que haga referencia a una plantilla de política existente y que especifique los valores de cualquier marcador de posición utilizado por la plantilla.

En el siguiente ejemplo, se crea una política vinculada a una plantilla que utiliza una plantilla con la siguiente instrucción:

```
permit(
    principal in ?principal,
    action == PhotoFlash::Action::"view",
    resource == PhotoFlash::Photo::"VacationPhoto94.jpg"
);
```

También utiliza el siguiente archivo `definition.txt` para proporcionar el valor del parámetro `definition`:

```
{
    "templateLinked": {
        "policyTemplateId": "PTEXAMPLEabcdefg111111",
        "principal": {
            "entityType": "PhotoFlash::User",
            "entityId": "alice"
        }
    }
}
```

La salida muestra tanto el recurso, que se obtiene de la plantilla, como la entidad principal, que se obtiene del parámetro de definición.

```
$ aws verifiedpermissions create-policy \
```

```
    --definition file://definition.txt
    --policy-store-id PSEXAMPLEabcdefg111111
{
    "createdDate": "2023-05-22T18:57:53.298278+00:00",
    "lastUpdatedDate": "2023-05-22T18:57:53.298278+00:00",
    "policyId": "TPEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "TEMPLATELINKED",
    "principal": {
        "entityId": "alice",
        "entityType": "PhotoFlash::User"
    },
    "resource": {
        "entityId": "VacationPhoto94.jpg",
        "entityType": "PhotoFlash::Photo"
    }
}
```

Edición de plantillas de política de permisos verificados de Amazon

Puede editar o actualizar las plantillas de políticas en Verified Permissions utilizando el Consola de administración de AWS AWS CLI, el o el AWS SDKs. Al editar una plantilla de políticas, se actualizarán automáticamente las políticas que estén vinculadas a la plantilla o se basen en ella, así que tenga cuidado al editar las plantillas de políticas y asegúrese de no introducir accidentalmente un cambio que interrumpa su solicitud.

Puede cambiar los siguientes elementos de una plantilla de política:

- Los `action` referenciados por la plantilla de política
- Una cláusula de condición, como `when` y `unless`

No puede cambiar los siguientes elementos de una plantilla de política. Para cambiar cualquiera de estos elementos, tendrá que eliminar y volver a crear la plantilla de política.

- El efecto de una plantilla de política a partir de `permit` o `forbid`
- Al que `principal` hace referencia una plantilla de política
- Al que `resource` hace referencia una plantilla de política

Consola de administración de AWS

Para editar sus plantillas de política

1. Abra la consola de permisos verificados. Elige tu almacén de políticas.
2. En el panel de navegación de la izquierda, elija Plantillas de política. La consola muestra todas las plantillas de política que creó en el almacén de políticas actual.
3. Pulse el botón de opción situado junto a una plantilla de política para ver los detalles de la plantilla de política, como cuándo se creó, se actualizó y su contenido.
4. Seleccione Editar para editar sus plantillas de política. Actualice la descripción y el cuerpo de la política según sea necesario y, a continuación, seleccione Actualizar la plantilla de política.
5. Para eliminar una plantilla de política, pulse el botón de opción situado junto a la plantilla de política y, a continuación, seleccione Eliminar. Pulse Aceptar para confirmar la eliminación de la plantilla de política.

AWS CLI

Para editar una plantilla de política

Puede crear una política estática mediante la UpdatePolicyoperación. En el siguiente ejemplo, la plantilla de política especificada se actualiza sustituyendo su cuerpo de política por una nueva política definida en un archivo.

Contenido del archivo `template1.txt`:

```
permit(
    principal in ?principal,
    action == Action::"view",
    resource in ?resource)
when {
    principal has department && principal.department == "research"
};
```

```
$ aws verifiedpermissions update-policy-template \
    --policy-template-id PTEXAMPLEabcdefg111111 \
    --description "My updated template description" \
    --statement file://template1.txt \
    --policy-store-id PSEXAMPLEabcdefg111111
{
```

```
    "createdDate": "2023-05-17T18:58:48.795411+00:00",
    "lastUpdatedDate": "2023-05-17T19:18:48.870209+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyTemplateId": "PTEXAMPLEabcdefg111111"
}
```

Para actualizar el nombre de una plantilla de política

Puede establecer o actualizar el nombre de una plantilla de política al actualizar una plantilla de política. El nombre debe ser único para todas las plantillas de políticas del almacén de políticas y debe ir precedido de`name/`. Si no incluye el campo de nombre en la solicitud de actualización, el nombre existente no cambiará. Para eliminar un nombre, configúrelo en una cadena vacía.

```
$ aws verifiedpermissions update-policy-template \
    --policy-template-id PTEXAMPLEabcdefg111111 \
    --statement file://template1.txt \
    --policy-store-id PSEXAMPLEabcdefg111111 \
    --name name/example-policy-template
{
    "createdDate": "2023-05-17T18:58:48.795411+00:00",
    "lastUpdatedDate": "2023-05-17T19:18:48.870209+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyTemplateId": "PTEXAMPLEabcdefg111111"
}
```

Ejemplos de políticas vinculadas a plantillas de permisos verificados de Amazon

Al crear un almacén de políticas en Verified Permissions mediante el método de almacén de políticas de muestra, el almacén de políticas se crea con políticas predefinidas, plantillas de políticas y un esquema para el proyecto de ejemplo que haya elegido. Los siguientes ejemplos de políticas vinculadas a plantillas de Verified Permissions se pueden utilizar con los almacenes de políticas de muestra y sus políticas, plantillas de políticas y esquemas respectivos.

PhotoFlash ejemplos

El siguiente ejemplo muestra cómo se puede crear una política vinculada a una plantilla que utilice la plantilla de política Conceder acceso limitado a las fotos no privadas compartidas con un usuario y una foto individuales.

> **Note**
>
> El lenguaje de las políticas de Cedar considera que una entidad está `in`. Por lo tanto, `principal in User::"Alice"` es equivalente a `principal == User::"Alice"`.

```
permit (
 principal in PhotoFlash::User::"Alice",
 action in PhotoFlash::Action::"SharePhotoLimitedAccess",
 resource in PhotoFlash::Photo::"VacationPhoto94.jpg"
 );
```

El siguiente ejemplo muestra cómo se puede crear una política vinculada a una plantilla que utilice la plantilla de política Conceder acceso limitado a las fotos compartidas no privadas con un usuario y un álbum individuales.

```
permit (
 principal in PhotoFlash::User::"Alice",
 action in PhotoFlash::Action::"SharePhotoLimitedAccess",
 resource in PhotoFlash::Album::"Italy2023"
 );
```

El siguiente ejemplo muestra cómo se puede crear una política vinculada a una plantilla que utilice la plantilla de política Conceder acceso limitado a las fotos no privadas compartidas con un grupo de amigos y una foto individual.

```
permit (
 principal in PhotoFlash::FriendGroup::"Jane::MySchoolFriends",
 action in PhotoFlash::Action::"SharePhotoLimitedAccess",
 resource in PhotoFlash::Photo::"VacationPhoto94.jpg"
 );
```

En el siguiente ejemplo, se muestra cómo se puede crear una política vinculada a una plantilla que utilice la plantilla de política Conceder acceso limitado a las fotos compartidas de carácter no privado con un grupo de amigos y un álbum.

```
permit (
 principal in PhotoFlash::FriendGroup::"Jane::MySchoolFriends",
 action in PhotoFlash::Action::"SharePhotoLimitedAccess",
```

```
 resource in PhotoFlash::Album::"Italy2023"
 );
```

En el siguiente ejemplo, se muestra cómo se puede crear una política vinculada a una plantilla que utilice la plantilla de política Conceder acceso total a las fotos compartidas no privadas con un grupo de amigos y una foto individual.

```
permit (
 principal in PhotoFlash::UserGroup::"Jane::MySchoolFriends",
 action in PhotoFlash::Action::"SharePhotoFullAccess",
 resource in PhotoFlash::Photo::"VacationPhoto94.jpg"
 );
```

En el siguiente ejemplo, se muestra cómo se puede crear una política vinculada a una plantilla que utilice la plantilla de política Bloquear a un usuario de una cuenta.

```
forbid(
 principal == PhotoFlash::User::"Bob",
 action,
 resource in PhotoFlash::Account::"Alice-account"
 );
```

DigitalPetStore ejemplos

El almacén de políticas de DigitalPetStore muestra no incluye ninguna plantilla de políticas. Para ver las políticas incluidas en el almacén de políticas, seleccione Políticas en el panel de navegación de la izquierda después de crear el almacén de políticas de DigitalPetStoremuestra.

TinyToDo ejemplos

El siguiente ejemplo muestra cómo se puede crear una política vinculada a una plantilla que utilice la plantilla de política que da acceso al espectador a un usuario individual y a una lista de tareas.

```
permit (
    principal == TinyTodo::User::"https://cognito-idp.us-east-1.amazonaws.com/us-
east-1_h2aKCU1ts|5ae0c4b1-6de8-4dff-b52e-158188686f31|bob",
    action in [TinyTodo::Action::"ReadList", TinyTodo::Action::"ListTasks"],
    resource == TinyTodo::List::"1"
);
```

El siguiente ejemplo muestra cómo se puede crear una política vinculada a una plantilla que utilice la plantilla de política que da acceso de editor a un usuario individual y a una lista de tareas.

```
permit (
    principal == TinyTodo::User::"https://cognito-idp.us-east-1.amazonaws.com/us-
east-1_h2aKCU1ts|5ae0c4b1-6de8-4dff-b52e-158188686f31|bob",
    action in [
        TinyTodo::Action::"ReadList",
        TinyTodo::Action::"UpdateList",
        TinyTodo::Action::"ListTasks",
        TinyTodo::Action::"CreateTask",
        TinyTodo::Action::"UpdateTask",
        TinyTodo::Action::"DeleteTask"
    ],
    resource == TinyTodo::List::"1"
);
```

Proteja sus aplicaciones con fuentes de identidad y tokens

Proteja sus aplicaciones rápidamente creando una fuente de identidad que represente a un proveedor de identidad externo (IdP) en Amazon Verified Permissions. Las fuentes de identidad proporcionan información de un usuario que se autenticó con un IdP que tiene una relación de confianza con su almacén de políticas. Cuando la aplicación realiza una solicitud de autorización con un token de una fuente de identidad, el almacén de políticas puede tomar decisiones de autorización a partir de las propiedades del usuario y los permisos de acceso. Puede añadir un grupo de usuarios de Amazon Cognito o un IdP de OpenID Connect (OIDC) personalizado como fuente de identidad.

Puede utilizar proveedores de identidad de OpenID Connect (OIDC) () con permisos verificadosIdPs. Su aplicación puede generar solicitudes de autorización con tokens web JSON (JWTs) generados por un proveedor de identidad compatible con la OIDC. La identidad de usuario del token se asigna al ID principal. Con los tokens de ID, Verified Permissions asigna las reclamaciones de atributos a los atributos principales. Con los tokens de acceso, estas afirmaciones se asignan al contexto. Con ambos tipos de token, puedes asignar una reclamación similar `groups` a un grupo principal y crear políticas que evalúen el control de acceso basado en funciones (RBAC).

> **Note**
>
> Verified Permissions toma decisiones de autorización en función de la información de un token de IdP, pero no interactúa directamente con el IdP de ninguna manera.

Para ver un step-by-step tutorial sobre cómo crear la lógica de autorización para Amazon API Gateway REST APIs mediante un grupo de Amazon Cognito usuarios o un proveedor de identidad OIDC, consulte Autorizar el uso de permisos verificados de API Gateway APIs Amazon con Amazon Cognito o traiga su propio proveedor de identidad en el AWS blog de seguridad.

Temas

- Elegir el proveedor de identidad correcto
- Trabajar con fuentes Amazon Cognito de identidad
- Trabajar con fuentes de identidad del OIDC

Elegir el proveedor de identidad correcto

Si bien los permisos verificados funcionan con varios IdPs, ten en cuenta lo siguiente a la hora de decidir cuál usar en tu aplicación:

Amazon Cognito Úselo cuando:

- Está creando nuevas aplicaciones sin una infraestructura de identidad existente
- Desea grupos AWS de usuarios gestionados con funciones de seguridad integradas
- Necesita la integración de un proveedor de identidad social
- ¿Quieres una gestión simplificada de los tokens

Utilice los proveedores de OIDC cuando:

- Tiene una infraestructura de identidad existente (Auth0, Okta, Azure AD)
- Debe mantener una administración de usuarios centralizada
- Tiene requisitos de conformidad específicos IdPs

Trabajar con fuentes Amazon Cognito de identidad

Verified Permissions trabaja en estrecha colaboración con los grupos de usuarios de Amazon Cognito. Amazon Cognito JWTs tienen una estructura predecible. Verified Permissions reconoce esta estructura y aprovecha al máximo la información que contiene. Por ejemplo, puede implementar un modelo de autorización de control de acceso basado en roles (RBAC) con tokens de identificación o de acceso.

Una nueva fuente de identidad de grupos de usuarios de Amazon Cognito requiere la siguiente información:

- El Región de AWS.
- El ID del grupo de usuarios.
- El tipo de entidad principal que desea asociar a su fuente de identidad, por ejemplo`MyCorp::User`.
- El tipo de entidad de grupo principal que desea asociar a su fuente de identidad, por ejemplo`MyCorp::UserGroup`.
- El cliente IDs de su grupo de usuarios al que desea autorizar para realizar solicitudes a su almacén de políticas.

Como los permisos verificados solo funcionan con grupos de usuarios de Amazon Cognito en la misma cuenta Cuenta de AWS, no puede especificar una fuente de identidad en otra cuenta. Verified Permissions establece el prefijo de la entidad (el identificador de la fuente de identidad al que debe hacer referencia en las políticas que actúan sobre los principios del grupo de usuarios) como el ID de su grupo de usuarios, por ejemplo. `us-west-2_EXAMPLE` En este caso, haría referencia a un usuario de ese grupo de usuarios con un ID como `a1b2c3d4-5678-90ab-cdef-EXAMPLE22222` `us-west-2_EXAMPLE|a1b2c3d4-5678-90ab-cdef-EXAMPLE22222`

Las notificaciones de los tokens del grupo de usuarios pueden contener atributos, ámbitos, grupos IDs, clientes y datos personalizados. Amazon Cognito JWTstienen la capacidad de incluir una variedad de información que puede contribuir a las decisiones de autorización en los permisos verificados. Entre ellos se incluyen:

1. El nombre de usuario y las reclamaciones grupales incluyen un `cognito:` prefijo
2. Atributos de usuario personalizados con un `custom: prefix`
3. Las reclamaciones personalizadas se añaden en tiempo de ejecución
4. El estándar de la OIDC afirma como y `sub` `email`

Tratamos estas reclamaciones en detalle y cómo gestionarlas en las políticas de permisos verificados, en. Asignación de Amazon Cognito tokens al esquema

> **Important**
>
> Si bien puedes revocar los Amazon Cognito tokens antes de que caduquen, se JWTs consideran recursos apátridas que son autónomos y están sujetos a una firma y validez. Por lo general, los servicios que cumplen con el RFC 7519 de JSON Web Token validan los tokens de forma remota y no están obligados a validarlos con el emisor. Esto significa que los permisos verificados pueden conceder acceso en función de un token que se haya revocado o emitido a un usuario que se haya eliminado posteriormente. Para reducir este riesgo, le recomendamos que cree tokens con una validez lo más corta posible y que revoque los tokens de actualización cuando desee eliminar la autorización para continuar con la sesión de un usuario. Para obtener más información, consulta Finalizar las sesiones de usuario con la revocación de un token

En el siguiente ejemplo, se muestra cómo puede crear una política que haga referencia a algunas de las reclamaciones del grupo de usuarios de Amazon Cognito asociadas a un principal.

```
permit(
     principal,
     action,
     resource == ExampleCo::Photo::"VacationPhoto94.jpg"
)
when {
     principal["cognito:username"]) == "alice" &&
     principal["custom:department"]) == "Finance"
};
```

En el siguiente ejemplo, se muestra cómo se puede crear una política que haga referencia a un principal que sea un usuario de un grupo de usuarios de Cognito. Tenga en cuenta que el ID principal adopta la forma de`"<userpool-id>|<sub>"`.

```
permit(
     principal == ExampleCo::User::"us-east-1_example|a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",
     action,
     resource == ExampleCo::Photo::"VacationPhoto94.jpg"
);
```

Las políticas de Cedar para las fuentes de identidad de los grupos de usuarios de Verified Permissions utilizan una sintaxis especial para los nombres de las notificaciones que contienen caracteres distintos de los alfanuméricos y los guiones bajos ()_. Esto incluye las notificaciones de prefijos de grupos de usuarios que contienen un `:` carácter, como y. `cognito:username` `custom:department` Para escribir una condición de la póliza que haga referencia a la `custom:department` afirmación `cognito:username` o a la reclamación, escríbalas como `principal["cognito:username"]` y`principal["custom:department"]`, respectivamente.

> **Note**
>
> Si un token contiene una reclamación con un `custom:` prefijo `cognito:` o y un nombre de reclamación con el valor literal `cognito` o`custom`, una solicitud de autorización con un prefijo no IsAuthorizedWithTokense aceptará con un`ValidationException`.

Para obtener más información sobre la representación cartográfica de las reclamaciones, consulteAsignación de Amazon Cognito tokens al esquema. Para obtener más información sobre

la autorización de Amazon Cognito los usuarios, consulte Autorización con permisos verificados de Amazon en la Guía para desarrolladores de Amazon Cognito.

Temas

- Creación de fuentes de Amazon Cognito identidad de Amazon Verified Permissions
- Edición de fuentes de Amazon Cognito identidad de Amazon Verified Permissions
- Asignación de Amazon Cognito tokens al esquema
- Validación de clientes y audiencias para Amazon Cognito

Creación de fuentes de Amazon Cognito identidad de Amazon Verified Permissions

El siguiente procedimiento agrega una fuente de identidad a un almacén de políticas existente.

También puede crear una fuente de identidad al crear un nuevo almacén de políticas en la consola de permisos verificados. En este proceso, puede importar automáticamente las notificaciones de los tokens de su fuente de identidad a los atributos de la entidad. Elige la opción Configuración guiada o Configuración con API Gateway un proveedor de identidad. Estas opciones también crean políticas iniciales.

> **Note**
>
> Las fuentes de identidad no están disponibles en el panel de navegación de la izquierda hasta que haya creado un almacén de políticas. Las fuentes de identidad que cree están asociadas al almacén de políticas actual.

Puede omitir el tipo de entidad principal al crear una fuente de identidad create-identity-sourceen la API de permisos verificados AWS CLI o CreateIdentitySourceen ella. Sin embargo, un tipo de entidad en blanco crea una fuente de identidad con un tipo de entidad de`AWS::Cognito`. El nombre de esta entidad no es compatible con el esquema del almacén de políticas. Para integrar Amazon Cognito las identidades en el esquema del almacén de políticas, debe establecer el tipo de entidad principal en una entidad de almacén de políticas compatible.

Consola de administración de AWS

Para crear una fuente de identidad de un grupo de usuarios de Amazon Cognito

1. Abra la consola de permisos verificados. Elige tu almacén de políticas.
2. En el panel de navegación de la izquierda, elija Fuentes de identidad.
3. Seleccione Crear fuente de identidad.
4. En Detalles del grupo de usuarios de Cognito, seleccione Región de AWS e introduzca el ID del grupo de usuarios para su fuente de identidad.
5. En Configuración principal, en Tipo principal, elija el tipo de entidad para los principales de esta fuente. Las identidades de los grupos de usuarios de Amazon Cognito conectados se asignarán al tipo de entidad principal seleccionado.
6. En Configuración de grupo, seleccione Usar el grupo de Cognito si quiere mapear la notificación del grupo `cognito:groups` de usuarios. Elija un tipo de entidad que sea principal del tipo principal.
7. En Validación de la aplicación del cliente, elija si desea validar la aplicación del cliente IDs.
   - Para validar la aplicación cliente IDs, elija Aceptar solo los tokens con una aplicación cliente coincidente IDs. Elija Agregar nuevo ID de aplicación cliente para cada ID de aplicación cliente que desee validar. Para eliminar un ID de aplicación cliente que se haya agregado, elija Eliminar junto al ID de la aplicación cliente.
   - Seleccione No validar la aplicación cliente IDs si no desea validar la aplicación cliente IDs.
8. Seleccione Crear fuente de identidad.
9. (Opcional) Si su almacén de políticas tiene un esquema, antes de poder hacer referencia a los atributos que extrae de los tokens de identidad o de acceso en sus políticas de Cedar, debe actualizar su esquema para que Cedar conozca el tipo de principal que crea su fuente de identidad. Esta incorporación al esquema debe incluir los atributos a los que desee hacer referencia en sus políticas de Cedar. Para obtener más información sobre cómo asignar los atributos de los Amazon Cognito tokens a los atributos principales de Cedar, consulteAsignación de Amazon Cognito tokens al esquema.

> **Note**
>
> Al crear un almacén de políticas vinculado a una API o utilizar Configurar con API Gateway un proveedor de identidades al crear almacenes de políticas, Verified

Permissions consulta los atributos de usuario del grupo de usuarios y crea un esquema en el que el tipo principal se rellena con los atributos del grupo de usuarios.

10. Cree políticas que utilicen la información de los tokens para tomar decisiones de autorización. Para obtener más información, consulte Creación de políticas estáticas de Amazon Verified Permissions.

Ahora que ha creado una fuente de identidad, actualizado el esquema y creado políticas, utilice Verified Permissions `IsAuthorizedWithToken` para tomar decisiones de autorización. Para obtener más información, consulta la guía IsAuthorizedWithTokende referencia de la API de permisos verificados de Amazon.

AWS CLI

Para crear una fuente de identidad de un grupo de usuarios de Amazon Cognito

Puede crear una fuente de identidad mediante la CreateIdentitySourceoperación. El siguiente ejemplo crea una fuente de identidad que puede acceder a las identidades autenticadas de un grupo de Amazon Cognito usuarios.

1. Cree un `config.txt` archivo que contenga los siguientes detalles del grupo de Amazon Cognito usuarios para que los utilice el `--configuration` parámetro del `create-identity-source` comando.

```
{
    "cognitoUserPoolConfiguration": {
        "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-
west-2_1a2b3c4d5",
        "clientIds":["a1b2c3d4e5f6g7h8i9j0kalbmc"],
        "groupConfiguration": {
              "groupEntityType": "MyCorp::UserGroup"
        }
    }
}
```

2. Ejecute el siguiente comando para crear una fuente de Amazon Cognito identidad.

```
$ aws verifiedpermissions create-identity-source \
    --configuration file://config.txt \
    --principal-entity-type "User" \
    --policy-store-id 123456789012
```

```
{
    "createdDate": "2023-05-19T20:30:28.214829+00:00",
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

3. (Opcional) Si su almacén de políticas tiene un esquema, antes de poder hacer referencia a los atributos que extrae de los tokens de identidad o acceso en sus políticas de Cedar, debe actualizar su esquema para que Cedar sepa el tipo de principal que crea su fuente de identidad. Esta incorporación al esquema debe incluir los atributos a los que desee hacer referencia en sus políticas de Cedar. Para obtener más información sobre cómo asignar los atributos de los Amazon Cognito tokens a los atributos principales de Cedar, consulteAsignación de Amazon Cognito tokens al esquema.

   > **Note**
   >
   > Al crear un almacén de políticas vinculado a una API o utilizar Configurar con API Gateway un proveedor de identidades al crear almacenes de políticas, Verified Permissions consulta los atributos de usuario del grupo de usuarios y crea un esquema en el que el tipo principal se rellena con los atributos del grupo de usuarios.

4. Cree políticas que utilicen la información de los tokens para tomar decisiones de autorización. Para obtener más información, consulte Creación de políticas estáticas de Amazon Verified Permissions.

Ahora que ha creado una fuente de identidad, actualizado el esquema y creado políticas, utilice Verified Permissions `IsAuthorizedWithToken` para tomar decisiones de autorización. Para obtener más información, consulta la guía IsAuthorizedWithTokende referencia de la API de permisos verificados de Amazon.

Para obtener más información sobre el uso de los tokens de acceso e identidad de Amazon Cognito para los usuarios autenticados en Verified Permissions, consulte Autorización con Amazon Verified Permissions en la Guía para desarrolladores de Amazon Cognito.

Edición de fuentes de Amazon Cognito identidad de Amazon Verified Permissions

Puede editar algunos parámetros de su fuente de identidad después de crearla. No puedes cambiar el tipo de fuente de identidad, tienes que eliminarla y crear una nueva para cambiarla a OIDC o Amazon Cognito a OIDC. Amazon Cognito Si el esquema del almacén de políticas coincide con los atributos de la fuente de identidad, tenga en cuenta que debe actualizar el esquema por separado para reflejar los cambios que realice en la fuente de identidad.

Consola de administración de AWS

Para actualizar una fuente Amazon Cognito de identidad

1. Abra la consola de permisos verificados. Elige tu almacén de políticas.
2. En el panel de navegación de la izquierda, elija Fuentes de identidad.
3. Seleccione el ID de la fuente de identidad que desee editar.
4. Elija Edit (Edición de).
5. En Detalles del grupo de usuarios de Cognito, seleccione Región de AWS y escriba el ID del grupo de usuarios para su fuente de identidad.
6. En Detalles principales, puede actualizar el tipo principal de la fuente de identidad. Las identidades de los grupos de usuarios de Amazon Cognito conectados se asignarán al tipo de entidad principal seleccionado.
7. En Configuración de grupo, seleccione Usar grupos de Cognito si quiere mapear la notificación del grupo `cognito:groups` de usuarios. Elija un tipo de entidad que sea principal del tipo principal.
8. En Validación de la aplicación del cliente, elija si desea validar la aplicación del cliente IDs.
   - Para validar la aplicación cliente IDs, elija Aceptar solo los tokens con una aplicación cliente coincidente IDs. Elija Agregar nuevo ID de aplicación cliente para cada ID de aplicación cliente que desee validar. Para eliminar un ID de aplicación cliente que se haya agregado, elija Eliminar junto al ID de la aplicación cliente.
   - Seleccione No validar la aplicación cliente IDs si no desea validar la aplicación cliente IDs.
9. Seleccione Save changes (Guardar cambios).
10. Si ha cambiado el tipo de entidad principal de la fuente de identidad, debe actualizar el esquema para que refleje correctamente el tipo de entidad principal actualizado.

Para eliminar una fuente de identidad, pulse el botón de opción situado junto a una fuente de identidad y, a continuación, elija Eliminar fuente de identidad. Escriba `delete` en el cuadro de texto y, a continuación, seleccione Eliminar fuente de identidad para confirmar la eliminación de la fuente de identidad.

AWS CLI

Para actualizar una fuente Amazon Cognito de identidad

Puede actualizar una fuente de identidad mediante la UpdateIdentitySourceoperación. En el siguiente ejemplo, se actualiza la fuente de identidad especificada para que utilice un grupo Amazon Cognito de usuarios diferente.

1. Cree un `config.txt` archivo que contenga los siguientes detalles del grupo de Amazon Cognito usuarios para que los utilice el `--configuration` parámetro del `update-identity-source` comando.

```
{
    "cognitoUserPoolConfiguration": {
        "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-
west-2_1a2b3c4d5",
        "clientIds":["a1b2c3d4e5f6g7h8i9j0kalbmc"],
        "groupConfiguration": {
              "groupEntityType": "MyCorp::UserGroup"
        }
    }
}
```

2. Ejecute el siguiente comando para actualizar una fuente de Amazon Cognito identidad.

```
$ aws verifiedpermissions update-identity-source \
    --update-configuration file://config.txt \
    --policy-store-id 123456789012
{
    "createdDate": "2023-05-19T20:30:28.214829+00:00",
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

> **Note**
>
> Si cambia el tipo de entidad principal de la fuente de identidad, debe actualizar el esquema para que refleje correctamente el tipo de entidad principal actualizado.

Asignación de Amazon Cognito tokens al esquema

Es posible que desee añadir una fuente de identidad a un almacén de políticas y asignar las reclamaciones de los proveedores, o símbolos, a su esquema de almacén de políticas. Puede automatizar este proceso mediante la configuración guiada para crear su almacén de políticas con una fuente de identidad o actualizar el esquema manualmente una vez creado el almacén de políticas. Una vez que haya asignado los tokens al esquema, puede crear políticas que hagan referencia a ellos.

Esta sección de la guía del usuario contiene la siguiente información:

- Cuándo puede rellenar automáticamente los atributos de un esquema de almacén de políticas
- ¿Cómo utilizar las notificaciones Amazon Cognito simbólicas en tus políticas de permisos verificados
- ¿Cómo crear manualmente un esquema para una fuente de identidad

Los almacenes de políticas vinculados a la API y los almacenes de políticas con una fuente de identidad que se crearon mediante una configuración guiada no requieren la asignación manual de los atributos del token de identidad (ID) al esquema. Puede proporcionar permisos verificados con los atributos de su grupo de usuarios y crear un esquema que se complete con los atributos de los usuarios. En la autorización del token de identificación, Verified Permissions asigna las reclamaciones a los atributos de una entidad principal. Es posible que tengas que asignar manualmente Amazon Cognito los tokens a tu esquema en las siguientes condiciones:

- Ha creado un almacén de políticas o un almacén de políticas vacío a partir de una muestra.
- Desea extender el uso de los tokens de acceso más allá del control de acceso basado en roles (RBAC).
- Los almacenes de políticas se crean con la API REST de permisos verificados, un AWS SDK o el. AWS CDK

Para Amazon Cognito utilizarlos como fuente de identidad en su almacén de políticas de permisos verificados, debe tener atributos de proveedor en su esquema. El esquema es fijo y debe corresponder a las entidades que crean los tokens del proveedor IsAuthorizedWithTokeno a las solicitudes de BatchIsAuthorizedWithTokenAPI. Si ha creado su almacén de políticas de forma que rellene automáticamente su esquema a partir de la información del proveedor en un token de identificación, está listo para escribir políticas. Si crea un almacén de políticas sin un esquema para su fuente de identidad, debe agregar atributos de proveedor al esquema que coincidan con las entidades creadas mediante las solicitudes de API. A continuación, puede escribir políticas utilizando los atributos del token del proveedor.

Para obtener más información sobre el uso del ID de Amazon Cognito y los tokens de acceso para los usuarios autenticados en los permisos verificados, consulte Autorización con permisos verificados de Amazon en la Guía para desarrolladores de Amazon Cognito.

Temas

- Asignación de los tokens de identificación al esquema
- Asignar tokens de acceso
- Notación alternativa para las reclamaciones Amazon Cognito delimitadas por dos puntos
- Lo que debe saber sobre el mapeo de esquemas

Asignación de los tokens de identificación al esquema

Verified Permissions procesa las reclamaciones de los tokens de identificación como atributos del usuario: sus nombres y cargos, su pertenencia a un grupo y su información de contacto. Los identificadores son especialmente útiles en un modelo de autorización de control de acceso basado en atributos (ABAC). Si quieres que los permisos verificados analicen el acceso a los recursos en función de quién realiza la solicitud, elige los tokens de identificación como fuente de identidad.

Amazon Cognito Los tokens de identificación funcionan con la mayoría de las bibliotecas de terceros dependientes del OIDC. Amplían las características del OIDC con afirmaciones adicionales. La aplicación puede autenticar al usuario con las operaciones de la API de autenticación de los grupos de usuarios de Amazon Cognito o con la interfaz de usuario alojada en el grupo de usuarios. Para obtener más información, consulte Uso de la API y los puntos de enlace en la Amazon Cognito Guía para desarrolladores.

Afirmaciones útiles en los Amazon Cognito identificadores

cognito:username y *preferred_username*

Variantes del nombre de usuario del usuario.

sub

El identificador de usuario único (UUID) del usuario

Reclamaciones con prefijo *custom:*

Un prefijo para los atributos personalizados del grupo de usuarios, como. `custom:employmentStoreCode`

Notificaciones estándar

El OIDC estándar afirma como y. `email` `phone_number` Para obtener más información, consulte las afirmaciones estándar de OpenID Connect Core 1.0 que incorporan el conjunto de erratas 2.

cognito:groups

Pertenencias a grupos de un usuario. En un modelo de autorización basado en el control de acceso basado en funciones (RBAC), esta afirmación presenta las funciones que puede evaluar en sus políticas.

Reclamaciones transitorias

Reclamaciones que no son propiedad del usuario, pero que se agregan en tiempo de ejecución mediante un disparador Lambda previo a la generación del token. Las afirmaciones transitorias se parecen a las afirmaciones estándar, pero están fuera de la norma, por ejemplo`tenant`, o. `department`

En las políticas que hacen referencia a Amazon Cognito atributos que tienen un `:` separador, haga referencia a los atributos en el formato`principal["cognito:username"]`. La afirmación de los roles `cognito:groups` es una excepción a esta regla. Verified Permissions asigna el contenido de esta declaración a las entidades principales de la entidad de usuario.

Para obtener más información sobre la estructura de los tokens de ID de los grupos de usuarios de Amazon Cognito, consulte Uso del token de ID en la Guía para Amazon Cognito desarrolladores.

El siguiente ejemplo de token de identificación tiene cada uno de los cuatro tipos de atributos. Incluye la afirmación Amazon Cognito específica`cognito:username`, la notificación

personalizadacustom:employmentStoreCode, la afirmación estándar y la reclamación email transitoria. tenant

```
{
    "sub": "91eb4550-XXX",
    "cognito:groups": [
        "Store-Owner-Role",
        "Customer"
    ],
    "email_verified": true,
    "clearance": "confidential",
    "iss": "https://cognito-idp.us-east-2.amazonaws.com/us-east-2_EXAMPLE",
    "cognito:username": "alice",
    "custom:employmentStoreCode": "petstore-dallas",
    "origin_jti": "5b9f50a3-05da-454a-8b99-b79c2349de77",
    "aud": "1example23456789",
    "event_id": "0ed5ad5c-7182-4ecf-XXX",
    "token_use": "id",
    "auth_time": 1687885407,
    "department": "engineering",
    "exp": 1687889006,
    "iat": 1687885407,
    "tenant": "x11app-tenant-1",
    "jti": "a1b2c3d4-e5f6-a1b2-c3d4-TOKEN1111111",
    "email": "alice@example.com"
}
```

Al crear una fuente de identidad con el grupo de Amazon Cognito usuarios, se especifica el tipo de entidad principal con la que Verified Permissions genera las solicitudes de autorización. IsAuthorizedWithToken Después, sus políticas pueden probar los atributos de esa entidad principal como parte de la evaluación de esa solicitud. Su esquema define el tipo y los atributos principales de una fuente de identidad y, a continuación, puede hacer referencia a ellos en sus políticas de Cedar.

También especifica el tipo de entidad de grupo que desea obtener de la afirmación del grupo de fichas de identificación. En las solicitudes de autorización, Verified Permissions asigna cada miembro de la reclamación del grupo a ese tipo de entidad de grupo. En las políticas, puede hacer referencia a esa entidad del grupo como principal.

El siguiente ejemplo muestra cómo reflejar los atributos del token de identidad de ejemplo en su esquema de Verified Permissions. Para obtener más información sobre cómo editar el esquema,

consulte Edición de esquemas de almacenamiento de políticas. Si la configuración de su fuente de identidad especifica el tipo de entidad principal `User`, puede incluir algo similar al siguiente ejemplo para que esos atributos estén disponibles para Cedar.

```
"User": {
   "shape": {
      "type": "Record",
      "attributes": {
         "cognito:username": {
            "type": "String",
            "required": false
         },
         "custom:employmentStoreCode": {
            "type": "String",
            "required": false
         },
         "email": {
            "type": "String"
         },
         "tenant": {
            "type": "String",
            "required": true
         }
      }
   }
}
```

Para ver un ejemplo de política que se validará con este esquema, consulteRefleja Amazon Cognito Atributos del token de identificación.

Asignar tokens de acceso

Verified Permissions procesa las notificaciones de token de acceso distintas de las declaradas por el grupo como atributos de la acción o atributos de contexto. Además de la pertenencia a un grupo, los tokens de acceso de su IdP pueden contener información sobre el acceso a la API. Los tokens de acceso son útiles en los modelos de autorización que utilizan el control de acceso basado en roles (RBAC). Los modelos de autorización que se basan en declaraciones de token de acceso distintas de la pertenencia a un grupo requieren un esfuerzo adicional en la configuración del esquema.

Amazon Cognito Los tokens de acceso tienen afirmaciones que se pueden utilizar para la autorización:

Afirmaciones útiles en los tokens de Amazon Cognito acceso

client_id

El ID de la aplicación cliente de una parte que confía en el OIDC. Con el ID de cliente, Verified Permissions puede comprobar que la solicitud de autorización proviene de un cliente autorizado para el almacén de políticas. En la autorización machine-to-machine (M2M), el sistema solicitante autoriza una solicitud con un secreto de cliente y proporciona el identificador del cliente y los alcances como prueba de la autorización.

scope

Los ámbitos OAuth 2.0 que representan los permisos de acceso del portador del token.

cognito:groups

Pertenencias a grupos de un usuario. En un modelo de autorización basado en el control de acceso basado en funciones (RBAC), esta afirmación presenta las funciones que puede evaluar en sus políticas.

Reclamaciones transitorias

Reclamaciones que no son un permiso de acceso, pero que se añaden en tiempo de ejecución mediante un disparador Lambda previo a la generación del token de un grupo de usuarios. Las afirmaciones transitorias se parecen a las afirmaciones estándar, pero están fuera del estándar, por ejemplo`tenant`, o. `department` La personalización de los tokens de acceso añade un coste a tu AWS factura.

Para obtener más información sobre la estructura de los tokens de acceso de los grupos de usuarios de Amazon Cognito, consulte Uso del token de acceso en la Guía para Amazon Cognito desarrolladores.

Un token de Amazon Cognito acceso se asigna a un objeto de contexto cuando se transfiere a Verified Permissions. Se puede hacer referencia a los atributos del token de acceso mediante `context.token.`*`attribute_name`*. El siguiente ejemplo de token de acceso incluye tanto el `client_id` como las notificaciones de `scope`.

```
{
    "sub": "91eb4550-9091-708c-a7a6-9758ef8b6b1e",
    "cognito:groups": [
        "Store-Owner-Role",
        "Customer"
```

```
    ],
    "iss": "https://cognito-idp.us-east-2.amazonaws.com/us-east-2_EXAMPLE",
    "client_id": "1example23456789",
    "origin_jti": "a1b2c3d4-e5f6-a1b2-c3d4-TOKEN1111111",
    "event_id": "bda909cb-3e29-4bb8-83e3-ce6808f49011",
    "token_use": "access",
    "scope": "MyAPI/mydata.write",
    "auth_time": 1688092966,
    "exp": 1688096566,
    "iat": 1688092966,
    "jti": "a1b2c3d4-e5f6-a1b2-c3d4-TOKEN2222222",
    "username": "alice"
}
```

El siguiente ejemplo muestra cómo reflejar los atributos del token de acceso de ejemplo en su esquema de Verified Permissions. Para obtener más información sobre cómo editar el esquema, consulte Edición de esquemas de almacenamiento de políticas.

```
{
   "MyApplication": {
      "actions": {
         "Read": {
            "appliesTo": {
               "context": {
                  "type": "ReusedContext"
               },
               "resourceTypes": [
                  "Application"
               ],
               "principalTypes": [
                  "User"
               ]
            }
         }
      },
      ...
      ...
      "commonTypes": {
         "ReusedContext": {
            "attributes": {
               "token": {
                  "type": "Record",
                  "attributes": {
```

```
                    "scope": {
                       "type": "Set",
                       "element": {
                          "type": "String"
                       }
                    },
                    "client_id": {
                       "type": "String"
                    }
                 }
              }
           },
           "type": "Record"
        }
     }
  }
}
```

Para ver un ejemplo de política que se validará con este esquema, consulteRefleja Amazon Cognito atributos del token de acceso.

Notación alternativa para las reclamaciones Amazon Cognito delimitadas por dos puntos

En el momento en que se lanzó Verified Permissions, el esquema recomendado para los símbolos establecía como, Amazon Cognito por ejemplo, estas cadenas delimitadas por puntos `cognito:groups` y las `custom:store` convertía para utilizar el `.` carácter como delimitador jerárquico. Este formato se denomina notación de puntos. Por ejemplo, una referencia a lo que `cognito:groups` pasó a figurar `principal.cognito.groups` en sus políticas. Aunque puede seguir utilizando este formato, le recomendamos que cree su esquema y sus políticas con una notación entre corchetes. En este formato, una referencia `cognito:groups` se convierte `principal["cognito:groups"]` en una referencia en sus políticas. Los esquemas generados automáticamente para los identificadores de los grupos de usuarios desde la consola de permisos verificados utilizan la notación entre corchetes.

Puede seguir utilizando la notación de puntos en los esquemas y políticas creados manualmente para las fuentes de identidad. Amazon Cognito No puede utilizar la notación de puntos `:` ni ningún otro carácter no alfanumérico en el esquema o las políticas de ningún otro tipo de IdP de OIDC.

Un esquema de notación de puntos anida cada instancia de un `:` carácter como elemento secundario de la frase `cognito` o frase `custom` inicial, como se muestra en el siguiente ejemplo:

```
"CognitoUser": {
   "shape": {
      "type": "Record",
      "attributes": {
         "cognito": {
            "type": "Record",
            "required": true,
            "attributes": {
               "username": {
                  "type": "String",
                  "required": true
               }
            }
         },
         "custom": {
            "type": "Record",
            "required": true,
            "attributes": {
               "employmentStoreCode": {
                  "type": "String",
                  "required": true
               }
            }
         },
         "email": {
            "type": "String"
         },
         "tenant": {
            "type": "String",
            "required": true
         }
      }
   }
}
```

Para ver un ejemplo de política que se validará con este esquema y utilizará la notación de puntos, consulteUsa la notación de puntos para hacer referencia a los atributos.

Lo que debe saber sobre el mapeo de esquemas

El mapeo de atributos difiere entre los tipos de token

En la autorización del token de acceso, Verified Permissions asigna las reclamaciones al contexto. En la autorización del token de identificación, Verified Permissions asigna las reclamaciones a los atributos principales. En el caso de los almacenes de políticas que cree en la consola de permisos verificados, solo los almacenes de políticas vacíos y de ejemplo no tienen una fuente de identidad y requieren que complete el esquema con los atributos del grupo de usuarios para la autorización del token de identificación. La autorización de los tokens de acceso se basa en el control de acceso basado en roles (RBAC) con las solicitudes de pertenencia a grupos y no asigna automáticamente otras solicitudes al esquema del almacén de políticas.

Los atributos de la fuente de identidad no son obligatorios

Al crear una fuente de identidad en la consola de permisos verificados, no se marca ningún atributo como obligatorio. Esto evita que las reclamaciones incumplidas provoquen errores de validación en las solicitudes de autorización. Puede establecer los atributos como obligatorios según sea necesario, pero deben estar presentes en todas las solicitudes de autorización.

El RBAC no requiere atributos en el esquema

Los esquemas de las fuentes de identidad dependen de las asociaciones de entidades que realice al agregar la fuente de identidad. Una fuente de identidad asigna una reclamación a un tipo de entidad de usuario y otra a un tipo de entidad de grupo. Estas asignaciones de entidades son el núcleo de una configuración de fuente de identidad. Con esta información mínima, puede escribir políticas que realicen acciones de autorización para usuarios específicos y grupos específicos de los que los usuarios puedan ser miembros, en un modelo de control de acceso basado en roles (RBAC). La adición de notificaciones de token al esquema amplía el ámbito de autorización del almacén de políticas. Los atributos de usuario de los tokens de identificación contienen información sobre los usuarios que puede contribuir a la autorización del control de acceso basado en atributos (ABAC). Los atributos de contexto de los tokens de acceso tienen información similar a los ámbitos OAuth 2.0 que pueden aportar información adicional sobre el control de acceso por parte del proveedor, pero requieren modificaciones adicionales en el esquema.

Las opciones Configurar con API Gateway y un proveedor de identidades y Configuración guiada de la consola de permisos verificados asignan reclamos de token de ID al esquema. Este no es el caso de las solicitudes de token de acceso. Para añadir notificaciones de token de acceso no grupales a tu esquema, debes editarlo en modo JSON y añadir los atributos commonTypes. Para obtener más información, consulte Asignar tokens de acceso.

Elige un tipo de token

La forma en que el almacén de políticas trabaja con la fuente de identidad depende de una decisión clave en la configuración de la fuente de identidad: si va a procesar los tokens de identificación o de acceso. Con un proveedor de Amazon Cognito identidades, puede elegir el tipo de token al crear un almacén de políticas vinculado a una API. Al crear un almacén de políticas vinculado a una API, debe elegir si desea configurar la autorización para los tokens de ID o de acceso. Esta información afecta a los atributos del esquema que Verified Permissions aplica a su almacén de políticas y a la sintaxis del autorizador Lambda para su API. API Gateway Especialmente si desea beneficiarse de la asignación automática de las solicitudes de token de identificación a los atributos de la consola de permisos verificados, decida con antelación qué tipo de token quiere procesar antes de crear su fuente de identidad. Cambiar el tipo de token requiere un esfuerzo considerable para refactorizar las políticas y el esquema. En los siguientes temas se describe el uso de los identificadores de acceso y de identificación en los almacenes de pólizas.

El analizador Cedar requiere corchetes para algunos caracteres

Las políticas suelen hacer referencia a los atributos del esquema en un formato como`principal.username`. En el caso de la mayoría de los caracteres no alfanuméricos`:`, como`.`, o `/` que puedan aparecer en los nombres de las notificaciones de los tokens, Verified Permissions no puede analizar un valor de condición como `principal.cognito:username` o. `context.ip-address` En su lugar, debe formatear estas condiciones con una notación entre corchetes en el formato `principal["cognito:username"]` o`context["ip-address"]`, respectivamente. El carácter de subrayado _ es un carácter válido en los nombres de las reclamaciones y es la única excepción no alfanumérica a este requisito.

Un ejemplo parcial de un esquema de un atributo principal de este tipo es el siguiente:

```
"User": {
   "shape": {
      "type": "Record",
      "attributes": {
         "cognito:username": {
            "type": "String",
            "required": true
         },
         "custom:employmentStoreCode": {
            "type": "String",
            "required": true,
         },
         "email": {
            "type": "String",
```

```
            "required": false
         }
      }
   }
}
```

Un ejemplo parcial de un esquema de un atributo de contexto de este tipo tiene el siguiente aspecto:

```
"GetOrder": {
   "memberOf": [],
   "appliesTo": {
      "resourceTypes": [
         "Order"
      ],
      "context": {
         "type": "Record",
         "attributes": {
            "ip-address": {
               "required": false,
               "type": "String"
            }
   }
   },
      "principalTypes": [
         "User"
      ]
   }
}
```

Para ver un ejemplo de política que se validará con este esquema, consulteUsa la notación entre corchetes para hacer referencia a los atributos del token.

Validación de clientes y audiencias para Amazon Cognito

Al añadir una fuente de identidad a un almacén de políticas, Verified Permissions tiene opciones de configuración que validan el ID de cliente y la audiencia de los identificadores y los identificadores de acceso, además de verificar la `token_use` afirmación. Esta validación se realiza durante el procesamiento de las solicitudes de `BatchIsAuthorizedWithToken` API `IsAuthorizedWithToken` y las solicitudes. El comportamiento difiere entre los identificadores y los identificadores de acceso Amazon Cognito y entre las fuentes de identidad del OIDC. Con los proveedores de grupos de usuarios de Amazon Cognito, Verified Permissions puede validar el

ID de cliente tanto en el identificador como en el token de acceso. Con los proveedores de OIDC, Verified Permissions puede validar el ID del cliente en los tokens de ID y la audiencia en los tokens de acceso.

Un ID de cliente es un identificador asociado a la instancia del proveedor de identidad que utiliza tu aplicación, por ejemplo. `1example23456789` Una audiencia es una ruta URL asociada a la parte de confianza prevista, o al destino, del token de acceso, por ejemplo`https://mytoken.example.com`. Cuando se utilizan tokens de acceso, la `aud` afirmación siempre se asocia a la audiencia.

Amazon Cognito Los identificadores tienen una `aud` declaración que contiene el ID del cliente de la aplicación. Los tokens de acceso tienen una `client_id` declaración que también contiene el ID de cliente de la aplicación.

Cuando ingresas uno o más valores para la validación de la aplicación cliente en tu fuente de identidad, Verified Permissions compara esta lista de ID de clientes de aplicaciones con la afirmación del token de ID o la `aud` afirmación del token `client_id` de acceso. Verified Permissions no valida la URL del público de una parte que depende para Amazon Cognito las fuentes de identidad.

Client-side autorización para los JWT

Es posible que desee procesar los tokens web JSON en su aplicación y transferir sus solicitudes a Verified Permissions sin utilizar una fuente de identidad de un almacén de políticas. Puedes extraer los atributos de tu entidad de un token web JSON (JWT) y analizarlos para convertirlos en permisos verificados.

En este ejemplo, se muestra cómo se pueden invocar permisos verificados desde una aplicación mediante un JWT.[1]

```
async function authorizeUsingJwtToken(jwtToken) {

    const payload = await verifier.verify(jwtToken);

    let principalEntity = {
        entityType: "PhotoFlash::User", // the application needs to fill in the
 relevant user type
        entityId: payload["sub"], // the application need to use the claim that
 represents the user-id
    };
    let resourceEntity = {
```

```
        entityType: "PhotoFlash::Photo", //the application needs to fill in the
 relevant resource type
        entityId: "jane_photo_123.jpg", // the application needs to fill in the
 relevant resource id
    };
    let action = {
        actionType: "PhotoFlash::Action", //the application needs to fill in the
 relevant action id
        actionId: "GetPhoto", //the application needs to fill in the relevant action
 type
    };
    let entities = {
        entityList: [],
    };
    entities.entityList.push(...getUserEntitiesFromToken(payload));
    let policyStoreId = "PSEXAMPLEabcdefg111111"; // set your own policy store id

    const authResult = await client
        .isAuthorized({
        policyStoreId: policyStoreId,
        principal: principalEntity,
        resource: resourceEntity,
        action: action,
        entities,
        })
        .promise();

    return authResult;

}

function getUserEntitiesFromToken(payload) {
  let attributes = {};
  let claimsNotPassedInEntities = ['aud', 'sub', 'exp', 'jti', 'iss'];
  Object.entries(payload).forEach(([key, value]) => {
    if (claimsNotPassedInEntities.includes(key)) {
        return;
    }
    if (Array.isArray(value)) {
      var attibuteItem = [];
      value.forEach((item) => {
        attibuteItem.push({
          string: item,
        });
```

```
      });
      attributes[key] = {
        set: attibuteItem,
      };
    } else if (typeof value === 'string') {
      attributes[key] = {
        string: value,
      }
    } else if (typeof value === 'bigint' || typeof value ==='number') {
        attributes[key] = {
            long: value,
          }
    } else if (typeof value === 'boolean') {
        attributes[key] = {
            boolean: value,
       }
    }

  });

  let entityItem = {
    attributes: attributes,
    identifier: {
      entityType: "PhotoFlash::User",
      entityId: payload["sub"], // the application needs to use the claim that
 represents the user-id
    }
  };
  return [entityItem];
}
```

[1] Este ejemplo de código usa la biblioteca aws-jwt-verify para verificar los JWT firmados por. OIDC-compatible IdPs

Trabajar con fuentes de identidad del OIDC

También puede configurar cualquier IdP de OpenID Connect (OIDC) compatible como fuente de identidad de un almacén de políticas. Los proveedores de OIDC son similares a los grupos de usuarios de Amazon Cognito: se JWTs producen como producto de la autenticación. Para añadir un proveedor de OIDC, debe proporcionar una URL del emisor

Una nueva fuente de identidad del OIDC requiere la siguiente información:

- La URL del emisor. Los permisos verificados deben poder detectar un `.well-known/openid-configuration` punto final en esta URL.
- Registros CNAME que no incluyen caracteres comodín. Por ejemplo, no se `a.example.com` puede asignar a. `*.example.net` Por el contrario, no se `*.example.com` puede mapear a. `a.example.net`
- El tipo de token que quieres usar en las solicitudes de autorización. En este caso, ha elegido el token de identidad.
- Por ejemplo, el tipo de entidad de usuario que desea asociar a su fuente de identidad`MyCorp::User`.
- El tipo de entidad de grupo que desea asociar a su fuente de identidad, por ejemplo`MyCorp::UserGroup`.
- Un ejemplo de token de ID o una definición de las afirmaciones del token de ID.
- El prefijo que desea aplicar a la entidad IDs de usuario y grupo. En la CLI y la API, puede elegir este prefijo. En los almacenes de políticas que se crean con la opción Configurar con API Gateway y un proveedor de identidades o la opción de configuración guiada, Verified Permissions asigna un prefijo del nombre del emisor menos`https://`, por ejemplo. `MyCorp::User::"auth.example.com|a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"`

Para obtener más información sobre el uso de las operaciones de la API para autorizar solicitudes de fuentes del OIDC, consulte. Operaciones de API disponibles para la autorización

En el siguiente ejemplo se muestra cómo se puede crear una política que permita el acceso a los informes de fin de año a los empleados del departamento de contabilidad que tengan una clasificación confidencial y no estén en una oficina satélite. Verified Permissions obtiene estos atributos de las afirmaciones que figuran en el token de identificación del director.

Tenga en cuenta que al hacer referencia a un grupo en el principal, debe utilizar el `in` operador para que la política se evalúe correctamente.

```
permit(
     principal in MyCorp::UserGroup::"MyOIDCProvider|Accounting",
     action,
     resource in MyCorp::Folder::"YearEnd2024"
) when {
     principal.jobClassification == "Confidential" &&
     !(principal.location like "SatelliteOffice*")
};
```

Temas

- Creación de fuentes de identidad OIDC de permisos verificados de Amazon
- Edición de las fuentes de identidad OIDC de Amazon Verified Permissions
- Asignación de tokens OIDC al esquema
- Validación de clientes y audiencias para proveedores de OIDC

Creación de fuentes de identidad OIDC de permisos verificados de Amazon

El siguiente procedimiento agrega una fuente de identidad a un almacén de políticas existente.

También puede crear una fuente de identidad al crear un nuevo almacén de políticas en la consola de permisos verificados. En este proceso, puede importar automáticamente las notificaciones de los tokens de su fuente de identidad a los atributos de la entidad. Elige la opción Configuración guiada o Configuración con API Gateway un proveedor de identidad. Estas opciones también crean políticas iniciales.

> **Note**
>
> Las fuentes de identidad no están disponibles en el panel de navegación de la izquierda hasta que haya creado un almacén de políticas. Las fuentes de identidad que cree están asociadas al almacén de políticas actual.

Puede omitir el tipo de entidad principal al crear una fuente de identidad create-identity-sourceen la API de permisos verificados AWS CLI o CreateIdentitySourceen ella. Sin embargo, un tipo de entidad en blanco crea una fuente de identidad con un tipo de entidad de`AWS::Cognito`. El nombre de esta entidad no es compatible con el esquema del almacén de políticas. Para integrar Amazon Cognito las identidades en el esquema del almacén de políticas, debe establecer el tipo de entidad principal en una entidad de almacén de políticas compatible.

Consola de administración de AWS

Para crear una fuente de identidad de OpenID Connect (OIDC)

1. Abra la consola de permisos verificados. Elige tu almacén de políticas.
2. En el panel de navegación de la izquierda, elija Fuentes de identidad.
3. Seleccione Crear fuente de identidad.

4. Elija un proveedor de OIDC externo.
5. En URL del emisor, introduzca la URL de su emisor de OIDC. Este es el punto final del servicio que proporciona, por ejemplo, el servidor de autorización, las claves de firma y otra información sobre su proveedor. `https://auth.example.com` La URL del emisor debe alojar un documento de detección del OIDC en. `/.well-known/openid-configuration`
6. En Tipo de token, elija el tipo de OIDC JWT que desea que envíe su solicitud de autorización. Para obtener más información, consulte Asignación de tokens OIDC al esquema.
7. En Map, las reclamaciones de token para esquematizar entidades, elija una entidad de usuario y una afirmación de usuario como fuente de identidad. La entidad de usuario es una entidad de su almacén de políticas a la que quiere hacer referencia a los usuarios de su proveedor de OIDC. La afirmación de usuario proviene, por lo general`sub`, de su ID o token de acceso que contiene el identificador único de la entidad que se va a evaluar. Las identidades del IdP OIDC conectado se asignarán al tipo principal seleccionado.
8. (Opcional) En las notificaciones de token de mapa para esquematizar entidades, elija una entidad de grupo y una afirmación de grupo como fuente de identidad. La entidad del grupo es la matriz de la entidad del usuario. Las reclamaciones grupales se asignan a esta entidad. La afirmación de grupo proviene, normalmente`groups`, de su ID o token de acceso y contiene una cadena, un JSON o una cadena delimitada por espacios de nombres de grupos de usuarios para la entidad que se va a evaluar. Las identidades del IdP OIDC conectado se asignarán al tipo principal seleccionado.
9. En la validación (opcional), introduzca el cliente IDs o público URLs que desee que su almacén de políticas acepte en las solicitudes de autorización, si las hubiera.
10. Seleccione Crear fuente de identidad.
11. (Opcional) Si su almacén de políticas tiene un esquema, antes de poder hacer referencia a los atributos que extrae de los tokens de identidad o acceso en sus políticas de Cedar, debe actualizar el esquema para que Cedar conozca el tipo de principal que crea su fuente de identidad. Esta incorporación al esquema debe incluir los atributos a los que desee hacer referencia en sus políticas de Cedar. Para obtener más información sobre cómo asignar los atributos del token OIDC a los atributos principales de Cedar, consulte. Asignación de tokens OIDC al esquema
12. Cree políticas que utilicen la información de los tokens para tomar decisiones de autorización. Para obtener más información, consulte Creación de políticas estáticas de Amazon Verified Permissions.

Ahora que ha creado una fuente de identidad, actualizado el esquema y creado políticas, utilice Verified Permissions `IsAuthorizedWithToken` para tomar decisiones de autorización. Para obtener más información, consulta la guía IsAuthorizedWithTokende referencia de la API de permisos verificados de Amazon.

AWS CLI

Para crear una fuente de identidad OIDC

Puede crear una fuente de identidad mediante la CreateIdentitySourceoperación. El siguiente ejemplo crea una fuente de identidad que puede acceder a las identidades autenticadas desde un proveedor de identidades (IdP) del OIDC.

1. Cree un `config.txt` archivo que contenga los siguientes detalles de un IdP OIDC para que lo utilice el `--configuration` parámetro del comando. `create-identity-source`

```
{
    "openIdConnectConfiguration": {
        "issuer": "https://auth.example.com",
        "tokenSelection": {
                "identityTokenOnly": {
                        "clientIds":["1example23456789"],
                        "principalIdClaim": "sub"
                },
        },
        "entityIdPrefix": "MyOIDCProvider",
        "groupConfiguration": {
              "groupClaim": "groups",
              "groupEntityType": "MyCorp::UserGroup"
        }
    }
}
```

2. Ejecute el siguiente comando para crear una fuente de identidad OIDC.

```
$ aws verifiedpermissions create-identity-source \
    --configuration file://config.txt \
    --principal-entity-type "User" \
    --policy-store-id 123456789012
{
    "createdDate": "2023-05-19T20:30:28.214829+00:00",
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",
```

```
    "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

3. (Opcional) Si su almacén de políticas tiene un esquema, antes de poder hacer referencia a los atributos que extrae de los tokens de identidad o acceso en sus políticas de Cedar, debe actualizar su esquema para que Cedar sepa el tipo de principal que crea su fuente de identidad. Esta incorporación al esquema debe incluir los atributos a los que desee hacer referencia en sus políticas de Cedar. Para obtener más información sobre cómo asignar los atributos del token OIDC a los atributos principales de Cedar, consulte. Asignación de tokens OIDC al esquema
4. Cree políticas que utilicen la información de los tokens para tomar decisiones de autorización. Para obtener más información, consulte Creación de políticas estáticas de Amazon Verified Permissions.

Ahora que ha creado una fuente de identidad, actualizado el esquema y creado políticas, utilice Verified Permissions `IsAuthorizedWithToken` para tomar decisiones de autorización. Para obtener más información, consulta la guía IsAuthorizedWithTokende referencia de la API de permisos verificados de Amazon.

Edición de las fuentes de identidad OIDC de Amazon Verified Permissions

Puede editar algunos parámetros de su fuente de identidad después de crearla. No puedes cambiar el tipo de fuente de identidad, tienes que eliminarla y crear una nueva para cambiarla a OIDC o Amazon Cognito a OIDC. Amazon Cognito Si el esquema del almacén de políticas coincide con los atributos de la fuente de identidad, tenga en cuenta que debe actualizar el esquema por separado para reflejar los cambios que realice en la fuente de identidad.

Consola de administración de AWS

Para actualizar una fuente de identidad del OIDC

1. Abra la consola de permisos verificados. Elige tu almacén de políticas.
2. En el panel de navegación de la izquierda, elija Fuentes de identidad.
3. Seleccione el ID de la fuente de identidad que desee editar.
4. Elija Edit (Edición de).
5. En los detalles del proveedor del OIDC, cambia la URL del emisor según sea necesario.

6. Si el token de mapa hace referencia a los atributos del esquema, cambie las asociaciones entre las afirmaciones de usuarios y grupos y los tipos de entidades del almacén de políticas, según sea necesario. Después de cambiar los tipos de entidad, debe actualizar las políticas y los atributos del esquema para aplicarlos a los nuevos tipos de entidad.
7. En la validación de audiencia, añade o elimina los valores de audiencia que quieras aplicar.
8. Seleccione Save changes (Guardar cambios).

Para eliminar una fuente de identidad, pulse el botón de opción situado junto a una fuente de identidad y, a continuación, elija Eliminar fuente de identidad. Escriba `delete` en el cuadro de texto y, a continuación, seleccione Eliminar fuente de identidad para confirmar la eliminación de la fuente de identidad.

AWS CLI

Para actualizar una fuente de identidad del OIDC

Puede actualizar una fuente de identidad mediante la UpdateIdentitySourceoperación. En el siguiente ejemplo, se actualiza la fuente de identidad especificada para que utilice un proveedor de OIDC diferente.

1. Cree un `config.txt` archivo que contenga los siguientes detalles de un IdP OIDC para que lo utilice el `--configuration` parámetro del comando. `update-identity-source`

```
{
    "openIdConnectConfiguration": {
        "issuer": "https://auth2.example.com",
        "tokenSelection": {
                "identityTokenOnly": {
                        "clientIds":["2example10111213"],
                        "principalIdClaim": "sub"
                },
        },
        "entityIdPrefix": "MyOIDCProvider",
        "groupConfiguration": {
              "groupClaim": "groups",
              "groupEntityType": "MyCorp::UserGroup"
        }
    }
}
```

2. Ejecute el siguiente comando para actualizar una fuente de identidad del OIDC.

```
$ aws verifiedpermissions update-identity-source \
    --update-configuration file://config.txt \
    --policy-store-id 123456789012
{
    "createdDate": "2023-05-19T20:30:28.214829+00:00",
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

> **Note**
>
> Si cambia el tipo de entidad principal de la fuente de identidad, debe actualizar el esquema para que refleje correctamente el tipo de entidad principal actualizado.

Asignación de tokens OIDC al esquema

Es posible que desee añadir una fuente de identidad a un almacén de políticas y asignar las reclamaciones de los proveedores, o símbolos, a su esquema de almacén de políticas. Puede automatizar este proceso mediante la configuración guiada para crear su almacén de políticas con una fuente de identidad o actualizar el esquema manualmente una vez creado el almacén de políticas. Una vez que haya asignado los tokens al esquema, puede crear políticas que hagan referencia a ellos.

Esta sección de la guía del usuario contiene la siguiente información:

- Cuándo puede rellenar automáticamente los atributos de un esquema de almacén de políticas
- Cómo crear manualmente un esquema para una fuente de identidad

Los almacenes de políticas vinculados a la API y los almacenes de políticas con una fuente de identidad que se crearon mediante una configuración guiada no requieren la asignación manual de los atributos del token de identidad (ID) al esquema. Puede proporcionar permisos verificados con los atributos de su grupo de usuarios y crear un esquema que se complete con los atributos de los usuarios. En la autorización de un token de identificación, Verified Permissions asigna las reclamaciones a los atributos de una entidad principal.

Para usar un proveedor de identidades (IdP) de OIDC como fuente de identidad en el almacén de políticas de permisos verificados, debe tener atributos de proveedor en su esquema. El esquema es fijo y debe corresponder a las entidades que crean los tokens del proveedor IsAuthorizedWithTokeno BatchIsAuthorizedWithTokena las solicitudes de API. Si ha creado su almacén de políticas de forma que rellene automáticamente su esquema a partir de la información del proveedor en un token de identificación, está listo para escribir políticas. Si crea un almacén de políticas sin un esquema para su fuente de identidad, debe agregar atributos de proveedor al esquema que coincidan con las entidades creadas mediante las solicitudes de API. A continuación, puede escribir políticas utilizando los atributos del token del proveedor.

Temas

- Asignación de los identificadores al esquema
- Asignar tokens de acceso
- Lo que debe saber sobre el mapeo de esquemas

Asignación de los identificadores al esquema

Verified Permissions procesa las reclamaciones de los tokens de identificación como atributos del usuario: sus nombres y cargos, su pertenencia a un grupo y su información de contacto. Los identificadores son especialmente útiles en un modelo de autorización de control de acceso basado en atributos (ABAC). Si quieres que los permisos verificados analicen el acceso a los recursos en función de quién realiza la solicitud, elige los tokens de identificación como fuente de identidad.

Trabajar con fichas de identificación de un proveedor de OIDC es prácticamente lo mismo que trabajar con fichas de Amazon Cognito identificación. La diferencia está en las afirmaciones. Su IdP puede presentar atributos OIDC estándar o tener un esquema personalizado. Al crear un nuevo almacén de políticas en la consola de permisos verificados, puede agregar una fuente de identidad OIDC con un token de ID de ejemplo, o puede asignar manualmente las notificaciones de los tokens a los atributos del usuario. Como Verified Permissions no conoce el esquema de atributos de tu IdP, debes proporcionar esta información.

Para obtener más información, consulte Crear almacenes de políticas de Verified Permissions.

El siguiente es un ejemplo de esquema para un almacén de políticas con una fuente de identidad OIDC.

```
"User": {
   "shape": {
```

```
      "type": "Record",
      "attributes": {
         "email": {
            "type": "String"
         },
         "email_verified": {
            "type": "Boolean"
         },
         "name": {
            "type": "String",
            "required": true
         },
         "phone_number": {
            "type": "String"
         },
         "phone_number_verified": {
            "type": "Boolean"
         }
      }
   }
}
```

Para ver un ejemplo de política que se validará con este esquema, consulte. Refleja los atributos del token de ID de OIDC

Asignar tokens de acceso

Verified Permissions procesa las notificaciones de token de acceso distintas de las declaradas por el grupo como atributos de la acción o atributos de contexto. Además de la pertenencia a un grupo, los tokens de acceso de su IdP pueden contener información sobre el acceso a la API. Los tokens de acceso son útiles en los modelos de autorización que utilizan el control de acceso basado en roles (RBAC). Los modelos de autorización que se basan en declaraciones de token de acceso distintas de la pertenencia a un grupo requieren un esfuerzo adicional en la configuración del esquema.

La mayoría de los tokens de acceso de proveedores de OIDC externos se alinean estrechamente con los tokens de acceso. Amazon Cognito Un token de acceso OIDC se asigna a un objeto de contexto cuando se transfiere a Verified Permissions. Se puede hacer referencia a los atributos del token de acceso mediante `context.token.`*`attribute_name`*. El siguiente ejemplo de token de acceso OIDC incluye ejemplos de afirmaciones base.

```
{
    "sub": "91eb4550-9091-708c-a7a6-9758ef8b6b1e",
```

```
    "groups": [
        "Store-Owner-Role",
        "Customer"
    ],
    "iss": "https://auth.example.com",
    "client_id": "1example23456789",
    "aud": "https://myapplication.example.com"
    "scope": "MyAPI-Read",
    "exp": 1688096566,
    "iat": 1688092966,
    "jti": "a1b2c3d4-e5f6-a1b2-c3d4-TOKEN2222222",
    "username": "alice"
}
```

El siguiente ejemplo muestra cómo reflejar los atributos del token de acceso de ejemplo en su esquema de Verified Permissions. Para obtener más información sobre cómo editar el esquema, consulte Edición de esquemas de almacenamiento de políticas.

```
{
   "MyApplication": {
      "actions": {
         "Read": {
            "appliesTo": {
               "context": {
                  "type": "ReusedContext"
               },
               "resourceTypes": [
                  "Application"
               ],
               "principalTypes": [
                  "User"
               ]
            }
         }
      },
      ...
      ...
      "commonTypes": {
         "ReusedContext": {
            "attributes": {
               "token": {
                  "type": "Record",
                  "attributes": {
```

```
                    "scope": {
                       "type": "Set",
                       "element": {
                          "type": "String"
                       }
                    },
                    "client_id": {
                       "type": "String"
                    }
                 }
              }
           },
           "type": "Record"
        }
     }
  }
}
```

Para ver un ejemplo de política que se validará con este esquema, consulte. Refleja los atributos del token de acceso de OIDC

Lo que debe saber sobre el mapeo de esquemas

El mapeo de atributos difiere entre los tipos de token

En la autorización del token de acceso, Verified Permissions asigna las reclamaciones al contexto. En la autorización del token de identificación, Verified Permissions asigna las reclamaciones a los atributos principales. En el caso de los almacenes de políticas que cree en la consola de permisos verificados, solo los almacenes de políticas vacíos y de ejemplo no tienen una fuente de identidad y requieren que complete el esquema con los atributos del grupo de usuarios para la autorización del token de identificación. La autorización de los tokens de acceso se basa en el control de acceso basado en roles (RBAC) con las solicitudes de pertenencia a grupos y no asigna automáticamente otras solicitudes al esquema del almacén de políticas.

Los atributos de la fuente de identidad no son obligatorios

Al crear una fuente de identidad en la consola de permisos verificados, no se marca ningún atributo como obligatorio. Esto evita que las reclamaciones incumplidas provoquen errores de validación en las solicitudes de autorización. Puede establecer los atributos como obligatorios según sea necesario, pero deben estar presentes en todas las solicitudes de autorización.

El RBAC no requiere atributos en el esquema

Los esquemas de las fuentes de identidad dependen de las asociaciones de entidades que realice al agregar la fuente de identidad. Una fuente de identidad asigna una reclamación a un tipo de entidad de usuario y otra a un tipo de entidad de grupo. Estas asignaciones de entidades son el núcleo de una configuración de fuente de identidad. Con esta información mínima, puede escribir políticas que realicen acciones de autorización para usuarios específicos y grupos específicos de los que los usuarios puedan ser miembros, en un modelo de control de acceso basado en roles (RBAC). La adición de notificaciones de token al esquema amplía el ámbito de autorización del almacén de políticas. Los atributos de usuario de los tokens de identificación contienen información sobre los usuarios que puede contribuir a la autorización del control de acceso basado en atributos (ABAC). Los atributos de contexto de los tokens de acceso tienen información similar a los ámbitos OAuth 2.0 que pueden aportar información adicional sobre el control de acceso por parte del proveedor, pero requieren modificaciones adicionales en el esquema.

Las opciones Configurar con API Gateway y un proveedor de identidades y Configuración guiada de la consola de permisos verificados asignan reclamos de token de ID al esquema. Este no es el caso de las solicitudes de token de acceso. Para añadir notificaciones de token de acceso no grupales a tu esquema, debes editarlo en modo JSON y añadir los atributos commonTypes. Para obtener más información, consulte Asignar tokens de acceso.

Los grupos de OIDC afirman que admite varios formatos

Al añadir un proveedor de OIDC, puede elegir el nombre del grupo reclamado en su ID o en los tokens de acceso que desee asignar a la membresía del grupo de un usuario en su almacén de políticas. Los permisos verificados reconocen las solicitudes de los grupos en los siguientes formatos:

1. Cadena sin espacios: `"groups": "MyGroup"`
2. Lista delimitada por espacios:. `"groups": "MyGroup1 MyGroup2 MyGroup3"` Cada cadena es un grupo.
3. Lista JSON (delimitada por comas): `"groups": ["MyGroup1", "MyGroup2", "MyGroup3"]`

> **Note**
>
> Los permisos verificados interpretan cada cadena de una reclamación de grupo separada por espacios como un grupo independiente. Para interpretar el nombre de un grupo con un carácter de espacio como un grupo único, sustituya o elimine el espacio de la afirmación. Por ejemplo, formatee un grupo denominado `My Group` como`MyGroup`.

Elija un tipo de token

La forma en que el almacén de políticas trabaja con la fuente de identidad depende de una decisión clave en la configuración de la fuente de identidad: si va a procesar los tokens de identificación o de acceso. Con un proveedor de OIDC, debes elegir un tipo de token al añadir la fuente de identidad. Puedes elegir un identificador o un token de acceso, y tu elección no permitirá que el tipo de token no elegido se procese en tu almacén de pólizas. Especialmente si quieres beneficiarte de la asignación automática de las solicitudes de token de identificación a los atributos de la consola de permisos verificados, decide con antelación qué tipo de token quieres procesar antes de crear tu fuente de identidad. Cambiar el tipo de token requiere un esfuerzo considerable para refactorizar las políticas y el esquema. En los siguientes temas se describe el uso de los identificadores de acceso y de identificación en los almacenes de pólizas.

El analizador Cedar requiere corchetes para algunos caracteres

Las políticas suelen hacer referencia a los atributos del esquema en un formato como`principal.username`. En el caso de la mayoría de los caracteres no alfanuméricos`:`, como`.`, o `/` que puedan aparecer en los nombres de las notificaciones de los tokens, Verified Permissions no puede analizar un valor de condición como `principal.cognito:username` o. `context.ip-address` En su lugar, debe formatear estas condiciones con una notación entre corchetes en el formato `principal["cognito:username"]` o`context["ip-address"]`, respectivamente. El carácter de subrayado _ es un carácter válido en los nombres de las reclamaciones y es la única excepción no alfanumérica a este requisito.

Un ejemplo parcial de un esquema de un atributo principal de este tipo es el siguiente:

```
"User": {
   "shape": {
      "type": "Record",
      "attributes": {
         "cognito:username": {
            "type": "String",
            "required": true
         },
         "custom:employmentStoreCode": {
            "type": "String",
            "required": true,
         },
         "email": {
            "type": "String",
```

```
            "required": false
         }
      }
   }
}
```

Un ejemplo parcial de un esquema de un atributo de contexto de este tipo tiene el siguiente aspecto:

```
"GetOrder": {
   "memberOf": [],
   "appliesTo": {
      "resourceTypes": [
         "Order"
      ],
      "context": {
         "type": "Record",
         "attributes": {
            "ip-address": {
               "required": false,
               "type": "String"
            }
   }
   },
      "principalTypes": [
         "User"
      ]
   }
}
```

Para ver un ejemplo de política que se validará con este esquema, consulteUsa la notación entre corchetes para hacer referencia a los atributos del token.

Validación de clientes y audiencias para proveedores de OIDC

Al añadir una fuente de identidad a un almacén de políticas, Verified Permissions tiene opciones de configuración que validan el ID de cliente y la audiencia de los identificadores y los identificadores de acceso. Esta validación se lleva a cabo durante el procesamiento de las solicitudes a la `BatchIsAuthorizedWithToken` API `IsAuthorizedWithToken` y las solicitudes de API. El comportamiento difiere entre los identificadores y los identificadores de acceso Amazon Cognito y entre las fuentes de identidad del OIDC. Con los proveedores de grupos de usuarios de Amazon Cognito, Verified Permissions puede validar el ID de cliente tanto en el identificador como en el token

de acceso. Con los proveedores de OIDC, Verified Permissions puede validar el ID del cliente en los tokens de ID y la audiencia en los tokens de acceso.

A diferencia de Amazon Cognito los tokens, los tokens OIDC no tienen una afirmación estándar que indique si un token es un token de identificación o un token de acceso. Como resultado, Verified Permissions no verifica el tipo de token de las fuentes de identidad del OIDC. Usted es responsable de transferir el tipo de token que su almacén de políticas está configurado para procesar en cada solicitud `IsAuthorizedWithToken` o `BatchIsAuthorizedWithToken` solicitud.

Un ID de cliente es un identificador asociado a la instancia del proveedor de identidad que utiliza tu aplicación, por ejemplo`1example23456789`. Una audiencia es una ruta URL asociada a la parte de confianza prevista, o al destino, del token de acceso, por ejemplo`https://mytoken.example.com`. Cuando se utilizan los tokens de acceso, la `aud` afirmación siempre se asocia a la audiencia.

Los tokens de ID de OIDC tienen una `aud` declaración que contiene los identificadores de los clientes, por ejemplo. `1example23456789`

Los tokens de acceso de OIDC tienen una `aud` afirmación que contiene la URL de audiencia del token, por ejemplo`https://myapplication.example.com`, y una `client_id` afirmación que contiene los ID de cliente, por ejemplo. `1example23456789`

Al configurar su almacén de políticas, introduzca uno o más valores para la validación de audiencia que el almacén de políticas utilice para validar la audiencia de un token.

- Tokens de identificación: Verified Permissions valida el ID de cliente comprobando que al menos un miembro de los ID de cliente de la `aud` reclamación coincide con un valor de validación de audiencia.
- Tokens de acceso: Verified Permissions valida la audiencia comprobando que la URL de la notificación coincide con `aud` un valor de validación de audiencia. Si no existe ninguna `aud` reclamación, se puede validar la audiencia mediante las `client_id` afirmaciones `cid` o. Consulta con tu proveedor de identidad cuál es la afirmación y el formato de audiencia correctos.

Client-side autorización para los JWT

Es posible que desee procesar los tokens web JSON en su aplicación y transferir sus solicitudes a Verified Permissions sin utilizar una fuente de identidad de un almacén de políticas. Puedes extraer los atributos de tu entidad de un token web JSON (JWT) y analizarlos para convertirlos en permisos verificados.

En este ejemplo, se muestra cómo se pueden invocar permisos verificados desde una aplicación mediante un JWT.[1]

```
async function authorizeUsingJwtToken(jwtToken) {

    const payload = await verifier.verify(jwtToken);

    let principalEntity = {
        entityType: "PhotoFlash::User", // the application needs to fill in the
 relevant user type
        entityId: payload["sub"], // the application need to use the claim that
 represents the user-id
    };
    let resourceEntity = {
        entityType: "PhotoFlash::Photo", //the application needs to fill in the
 relevant resource type
        entityId: "jane_photo_123.jpg", // the application needs to fill in the
 relevant resource id
    };
    let action = {
        actionType: "PhotoFlash::Action", //the application needs to fill in the
 relevant action id
        actionId: "GetPhoto", //the application needs to fill in the relevant action
 type
    };
    let entities = {
        entityList: [],
    };
    entities.entityList.push(...getUserEntitiesFromToken(payload));
    let policyStoreId = "PSEXAMPLEabcdefg111111"; // set your own policy store id

    const authResult = await client
        .isAuthorized({
        policyStoreId: policyStoreId,
        principal: principalEntity,
        resource: resourceEntity,
        action: action,
        entities,
        })
        .promise();

    return authResult;
```

```
}

function getUserEntitiesFromToken(payload) {
  let attributes = {};
  let claimsNotPassedInEntities = ['aud', 'sub', 'exp', 'jti', 'iss'];
  Object.entries(payload).forEach(([key, value]) => {
    if (claimsNotPassedInEntities.includes(key)) {
        return;
    }
    if (Array.isArray(value)) {
      var attibuteItem = [];
      value.forEach((item) => {
        attibuteItem.push({
          string: item,
        });
      });
      attributes[key] = {
        set: attibuteItem,
      };
    } else if (typeof value === 'string') {
      attributes[key] = {
        string: value,
      }
    } else if (typeof value === 'bigint' || typeof value ==='number') {
        attributes[key] = {
            long: value,
          }
    } else if (typeof value === 'boolean') {
        attributes[key] = {
            boolean: value,
       }
    }

  });

  let entityItem = {
    attributes: attributes,
    identifier: {
      entityType: "PhotoFlash::User",
      entityId: payload["sub"], // the application needs to use the claim that
 represents the user-id
    }
  };
  return [entityItem];
```

```
}
```

[1] Este ejemplo de código usa la biblioteca aws-jwt-verify para verificar los JWT firmados por. OIDC-compatible IdPs

Integraciones para permisos verificados de Amazon

Las integraciones de Amazon Verified Permissions le ayudan a implementar una autorización detallada en sus aplicaciones, a la vez que minimizan el código y siguen las prácticas recomendadas específicas del marco. Estas integraciones proporcionan componentes y utilidades de middleware que conectan sin problemas su aplicación con los permisos verificados.

Con las integraciones, puede:

- Implemente la autorización en cuestión de minutos
- Siga los patrones y convenciones específicos del marco
- Reduzca los gastos de mantenimiento
- Minimice los posibles errores de implementación de la seguridad
- Céntrese en la lógica empresarial más que en el código de autorización

Cuando se agregan a la aplicación, las integraciones hacen lo siguiente:

1. Intercepte las solicitudes entrantes a través de un middleware específico del marco
2. Extraiga el contexto de autorización relevante de las solicitudes
3. Determine las decisiones de autorización mediante permisos verificados
4. Aplique el control de acceso en función de los resultados de la autorización

En la actualidad, Verified Permissions es compatible con los siguientes marcos:

- Express.js para aplicaciones Node.js

Integración de Express con los permisos verificados de Amazon

La integración de Verified Permissions Express proporciona un enfoque basado en el middleware para implementar la autorización en sus aplicaciones de Express.js. Con esta integración, puede proteger los puntos finales de su API mediante políticas de autorización detalladas sin modificar sus controladores de rutas existentes. La integración gestiona las comprobaciones de autorización de forma automática al interceptar las solicitudes, evaluarlas en función de las políticas definidas y garantizar que solo los usuarios autorizados puedan acceder a los recursos protegidos.

En este tema se explica cómo configurar la integración de Express, desde la creación de un almacén de políticas hasta la implementación y las pruebas del middleware de autorización. Si sigue estos pasos, puede añadir controles de autorización sólidos a su aplicación Express con cambios de código mínimos.

En este tema se hace referencia a los siguientes GitHub repositorios:

- cedar-policy/ authorization-for-expressjs: el middleware de autorización de Cedar para Express.js
- authorization-clients-jsverifiedpermissions/: los clientes de autorización de permisos verificados para JavaScript
- verifiedpermissions/examples/express-petstore: ejemplo de implementación con el middleware Express.js

Requisitos previos

Antes de implementar la integración de Express, asegúrese de tener:

- Una AWS cuenta con acceso a permisos verificados
- Node.js y npm instalados
- Una aplicación Express.js
- Un proveedor de identidad OpenID Connect (OIDC) (como) Amazon Cognito
- AWS CLIconfigurado con los permisos adecuados

Configuración de la integración

Paso 1: Crear un almacén de políticas

Cree un almacén de políticas mediante AWS CLI:

```
aws verifiedpermissions create-policy-store --validation-settings "mode=STRICT"
```

> **Note**
>
> Guarde el ID del almacén de políticas devuelto en la respuesta para usarlo en los pasos siguientes.

Paso 2: Instalar las dependencias

Instale los paquetes necesarios en su aplicación Express:

```
npm i --save @verifiedpermissions/authorization-clients-js
npm i --save @cedar-policy/authorization-for-expressjs
```

Configuración de la autorización

Paso 1: Genere y cargue el esquema de Cedar

Un esquema define el modelo de autorización de una aplicación, incluidos los tipos de entidades de la aplicación y las acciones que los usuarios pueden realizar. Se recomienda definir un espacio de nombres para el esquema. En este ejemplo, usaremos `YourNamespace`. Adjunta el esquema a los almacenes de políticas de permisos verificados y, cuando se agregan o modifican políticas, el servicio las valida automáticamente para compararlas con el esquema.

El `@cedar-policy/authorization-for-expressjs` paquete puede analizar las especificaciones de OpenAPI de su aplicación y generar un esquema de Cedar. Específicamente, el objeto paths es obligatorio en su especificación.

Si no tiene una especificación de OpenAPI, puede seguir las instrucciones rápidas del express-openapi-generatorpaquete para generar una especificación de OpenAPI.

Genere un esquema a partir de su especificación de OpenAPI:

```
npx @cedar-policy/authorization-for-expressjs generate-schema --api-spec schemas/
openapi.json --namespace YourNamespace --mapping-type SimpleRest
```

A continuación, formatee el esquema de Cedar para usarlo con. AWS CLI Para obtener más información sobre el formato específico requerido, consulteEsquema del almacén de políticas. Si necesitas ayuda para formatear el esquema, hay un script llamado `prepare-cedar-schema.sh` en el repositorio GitHubverifiedpermissions/examples. El siguiente es un ejemplo de llamada a ese script que genera el esquema formateado de permisos verificados en el archivo. `v2.cedarschema.forAVP.json`

```
./scripts/prepare-cedar-schema.sh v2.cedarschema.json v2.cedarschema.forAVP.json
```

Cargue el esquema formateado a su almacén de políticas y sustitúyalo por su `policy-store-id` ID de almacén de políticas:

```
aws verifiedpermissions put-schema \
  --definition file://v2.cedarschema.forAVP.json \
  --policy-store-id policy-store-id
```

Paso 2: Crear políticas de autorización

Si no se configura ninguna política, Cedar deniega todas las solicitudes de autorización. La integración del marco Express ayuda a iniciar este proceso mediante la generación de políticas de ejemplo basadas en el esquema generado anteriormente.

Cuando utilice esta integración en sus aplicaciones de producción, le recomendamos que cree nuevas políticas utilizando herramientas de infraestructura como código (IaaC). Para obtener más información, consulte Trabajando con AWS CloudFormation.

Genere ejemplos de políticas de Cedar:

```
npx @cedar-policy/authorization-for-expressjs generate-policies --schema
 v2.cedarschema.json
```

Esto generará ejemplos de políticas en el `/policies` directorio. A continuación, puede personalizar estas políticas en función de sus casos de uso. Por ejemplo:

```
// Defines permitted administrator user group actions
permit (
    principal in YourNamespace::UserGroup::"<userPoolId>|administrator",
    action,
    resource
);

// Defines permitted employee user group actions
permit (
    principal in YourNamespace::UserGroup::"<userPoolId>|employee",
    action in
        [YourNamespace::Action::"GET /resources",
         YourNamespace::Action::"POST /resources",
         YourNamespace::Action::"GET /resources/{resourceId}",
         YourNamespace::Action::"PUT /resources/{resourceId}"],
```

```
    resource
);
```

Formatee las políticas para usarlas con AWS CLI. Para obtener más información sobre el formato requerido, consulte create-policy en la AWS CLI referencia. Si necesitas ayuda para formatear las políticas, hay un script llamado `convert_cedar_policies.sh` en el repositorio GitHubverifiedpermissions/examples. La siguiente es una llamada a ese script:

```
./scripts/convert_cedar_policies.sh
```

Sube las políticas formateadas a Verified Permissions y `policy_1.json` sustitúyelas por la ruta y el nombre del archivo de políticas y `policy-store-id` por el ID del almacén de políticas:

```
aws verifiedpermissions create-policy \
  --definition file://policies/json/policy_1.json \
  --policy-store-id policy-store-id
```

Paso 3: Conectar un proveedor de identidad

De forma predeterminada, el middleware del autorizador de permisos verificados lee un token web JSON (JWT) incluido en el encabezado de autorización de la solicitud de API para obtener información del usuario. Los permisos verificados pueden validar el token además de realizar una evaluación de la política de autorización.

Cree un archivo de configuración de origen de identidad con un nombre `identity-source-configuration.txt` similar al siguiente con su `userPoolArn` manoclientId:

```
{
    "cognitoUserPoolConfiguration": {
        "userPoolArn": "arn:aws:cognito-idp:region:account:userpool/pool-id",
        "clientIds": ["client-id"],
        "groupConfiguration": {
            "groupEntityType": "YourNamespace::UserGroup"
        }
    }
}
```

Cree la fuente de identidad ejecutando el siguiente AWS CLI comando, `policy-store-id` sustituyéndolo por su ID de almacén de políticas:

```
aws verifiedpermissions create-identity-source \
  --configuration file://identity-source-configuration.txt \
  --policy-store-id policy-store-id \
  --principal-entity-type YourNamespace::User
```

Implementación del middleware de autorización

Actualice su aplicación Express para incluir el middleware de autorización. En este ejemplo, utilizamos identificadores de identidad, pero usted también puede utilizar identificadores de acceso. Para obtener más información, consulte authorization-for-expressjsenGitHub.

```
const { ExpressAuthorizationMiddleware } = require('@cedar-policy/authorization-for-
expressjs');

const { AVPAuthorizationEngine } = require('@verifiedpermissions/authorization-
clients');

const avpAuthorizationEngine = new AVPAuthorizationEngine({
    policyStoreId: 'policy-store-id',
    callType: 'identityToken'
});

const expressAuthorization = new ExpressAuthorizationMiddleware({
    schema: {
        type: 'jsonString',
        schema: fs.readFileSync(path.join(__dirname, '../v4.cedarschema.json'),
 'utf8'),
    },
    authorizationEngine: avpAuthorizationEngine,
    principalConfiguration: { type: 'identityToken' },
    skippedEndpoints: [],
    logger: {
        debug: (s) => console.log(s),
        log: (s) => console.log(s),
    }
});

// Add the middleware to your Express application
app.use(expressAuthorization.middleware);
```

Probando la integración

Para probar la implementación de la autorización, realiza solicitudes a los puntos finales de la API con diferentes tokens de usuario. El middleware de autorización evaluará automáticamente cada solicitud en función de las políticas que haya definido.

Por ejemplo, si ha configurado diferentes grupos de usuarios con diferentes permisos:

- Administradores: acceso completo a todos los recursos y funciones de administración
- Empleados: pueden ver, crear y actualizar los recursos
- Clientes: solo pueden ver los recursos

Puede validar que las políticas de permisos funcionan según lo esperado iniciando sesión con diferentes usuarios e intentando realizar varias operaciones. En la terminal de la aplicación Express, puede ver el resultado del registro que proporciona detalles adicionales sobre las decisiones de autorización.

Resolución de problemas

Si tiene errores de autorización, intente lo siguiente:

- Compruebe que el ID del almacén de políticas sea correcto
- Asegúrese de que su fuente de identidad esté configurada correctamente
- Compruebe que sus políticas tengan el formato correcto
- Valide que sus tokens JWT sean válidos

Siguientes pasos

Tras implementar la integración básica, considere lo siguiente:

- Implementar mapeadores personalizados para escenarios de autorización específicos
- Configurar la supervisión y el registro de las decisiones de autorización
- Crear políticas adicionales para diferentes roles de usuario

Implementación de la autorización en Amazon Verified Permissions

Tras crear el almacén de políticas, las políticas, las plantillas, el esquema y el modelo de autorización, estará listo para empezar a autorizar las solicitudes mediante los permisos verificados de Amazon. Para implementar la autorización de permisos verificados, debe combinar la configuración de las políticas de autorización AWS con la integración en una aplicación. Para integrar los permisos verificados en su aplicación, añada un AWS SDK e implemente los métodos que invocan la API de permisos verificados y generan decisiones de autorización en función de su almacén de políticas.

La autorización con permisos verificados es útil para los permisos de experiencia de usuario y los permisos de API en sus aplicaciones.

Permisos de UX

> Controle el acceso de los usuarios a la UX de su aplicación. Puede permitir que un usuario vea solo los formularios, botones, gráficos y otros recursos exactos a los que necesita acceder. Por ejemplo, cuando un usuario inicia sesión, es posible que desee determinar si el botón «Transferir fondos» está visible en su cuenta. También puedes controlar las acciones que puede realizar un usuario. Por ejemplo, en la misma aplicación bancaria, es posible que desee determinar si su usuario puede cambiar la categoría de una transacción.

Permisos de la API

> Controle el acceso de los usuarios a los datos. Las aplicaciones suelen formar parte de un sistema distribuido y reciben información de fuentes externas APIs. En el ejemplo de la aplicación bancaria en la que los permisos verificados permiten mostrar el botón «Transferir fondos», se debe tomar una decisión de autorización más compleja cuando el usuario inicia una transferencia. Los permisos verificados pueden autorizar la solicitud de API en la que se indican las cuentas de destino que son destinatarios de transferencias aptas y, a continuación, la solicitud para transferir la transferencia a la otra cuenta.

Los ejemplos que ilustran este contenido provienen de un ejemplo de almacén de políticas. Para continuar, cree el almacén de políticas de DigitalPetStoremuestra en su entorno de pruebas.

Para ver un ejemplo de aplicación integral que implementa permisos de experiencia de usuario mediante la autorización por lotes, consulte Utilizar permisos verificados de Amazon para obtener una autorización detallada a gran escala en el AWS blog de seguridad.

Temas

- Operaciones de API disponibles para la autorización
- Probar su modelo de autorización
- Integración de sus modelos de autorización con las aplicaciones

Operaciones de API disponibles para la autorización

La API de permisos verificados tiene las siguientes operaciones de autorización.

IsAuthorized

La operación de la `IsAuthorized` API es el punto de entrada a las solicitudes de autorización con permisos verificados. Debe enviar los elementos principales, de acción, de recursos, de contexto y de entidad. Verified Permissions evalúa su solicitud comparándola con todas las políticas del almacén de políticas solicitado que se aplican a las entidades de la solicitud.

IsAuthorizedWithToken

La `IsAuthorizedWithToken` operación genera una solicitud de autorización a partir de los datos del usuario en los tokens web JSON (JWTs). Verified Permissions funciona directamente con los proveedores de OIDC, por ejemplo Amazon Cognito , como fuente de identidad en tu almacén de políticas. Verified Permissions rellena todos los atributos del principal de tu solicitud a partir de las afirmaciones que figuran en la identificación de los usuarios o en los tokens de acceso. Puedes autorizar acciones y recursos a partir de los atributos de usuario o la pertenencia a un grupo en una fuente de identidad.

No puedes incluir información sobre los tipos principales de grupos o usuarios en una `IsAuthorizedWithToken` solicitud. Debe rellenar todos los datos principales en el JWT que proporcione.

BatchIsAuthorized

La `BatchIsAuthorized` operación procesa varias decisiones de autorización para un único principal o recurso en una sola solicitud de API. Esta operación agrupa las solicitudes en una sola operación por lotes que minimiza el uso de la cuota y devuelve las decisiones de autorización

para cada una de las 30 acciones anidadas complejas. Con la autorización por lotes para un único recurso, puede filtrar las acciones que un usuario puede realizar en un recurso. Con la autorización por lotes para un único principal, puede filtrar los recursos sobre los que un usuario puede realizar acciones.

BatchIsAuthorizedWithToken

La `BatchIsAuthorizedWithToken` operación procesa varias decisiones de autorización para un único principal en una solicitud de API. El principal lo proporciona la fuente de identidad del almacén de políticas en un identificador o token de acceso. Esta operación agrupa las solicitudes en una sola operación por lotes que minimiza el uso de la cuota y devuelve las decisiones de autorización para cada una de las 30 solicitudes de acciones y recursos como máximo. En sus políticas, puede autorizar su acceso desde sus atributos o su pertenencia a un grupo en un directorio de usuarios.

Del mismo `IsAuthorizedWithToken` modo, no puedes incluir información sobre los principales tipos de grupos o usuarios en una `BatchIsAuthorizedWithToken` solicitud. Debe rellenar todos los datos principales en el JWT que proporcione.

Probar su modelo de autorización

Para comprender el efecto de la decisión de autorización de permisos verificados de Amazon al implementar su aplicación, puede evaluar sus políticas a medida que las desarrolla con la API REST HTTPS Uso del banco de pruebas de permisos verificados de Amazon y con las solicitudes de la API REST de HTTPS a Verified Permissions. El banco de pruebas es una herramienta que sirve Consola de administración de AWS para evaluar las solicitudes de autorización y las respuestas en su almacén de políticas.

La API REST de permisos verificados es el siguiente paso en su desarrollo, a medida que pasa de la comprensión conceptual al diseño de la aplicación. La API de permisos verificados acepta solicitudes de autorización con IsAuthorizedy BatchIsAuthorizedcomo solicitudes de AWS API firmadas para puntos finales de servicios regionales. IsAuthorizedWithToken Para probar tu modelo de autorización, puedes generar solicitudes con cualquier cliente de API y comprobar que tus políticas devuelven las decisiones de autorización según lo previsto.

Por ejemplo, puede realizar una prueba `IsAuthorized` en un almacén de políticas de muestra con el siguiente procedimiento.

Test bench

1. Abra la consola de permisos verificados en la consola de permisos verificados. Cree un almacén de políticas a partir del almacén de políticas de muestra con el nombre DigitalPetStore.
2. Selecciona Test bench en tu nuevo almacén de políticas.
3. Rellena tu formulario de solicitud de banco de pruebas IsAuthorizeden la referencia de la API de permisos verificados. Los siguientes detalles reproducen las condiciones del ejemplo 4 que hace referencia a la DigitalPetStoremuestra.
    a. Pon a Alice como la directora. Para que el director tome medidas, elige `DigitalPetStore::User` e ingresa`Alice`.
    b. Establece el rol de Alice como cliente. Elija Agregar un padre`DigitalPetStore::Role`, elija e introduzca Cliente.
    c. Establezca el recurso como pedido «1234». Elija `DigitalPetStore::Order` e introduzca el recurso sobre el que actúa el principal`1234`.
    d. El `DigitalPetStore::Order` recurso requiere un `owner` atributo. Establece a Alice como la propietaria del pedido. Elige `DigitalPetStore::User` e ingresa `Alice`
    e. Alice solicitó ver el pedido. Para la acción que está tomando el director, elija`DigitalPetStore::Action::"GetOrder"`.
4. Elija Ejecutar solicitud de autorización. En un almacén de políticas sin modificar, esta solicitud da lugar a una `ALLOW` decisión. Tenga en cuenta la política de satisfacción que dio lugar a la decisión.
5. Elija Políticas en la barra de navegación izquierda. Revise la política estática con la descripción Customer Role: Get Order.
6. Observe que los permisos verificados permitieron la solicitud porque el principal tenía un rol de cliente y era el propietario del recurso.

REST API

1. Abra la consola de permisos verificados en la consola de permisos verificados. Cree un almacén de políticas a partir del almacén de políticas de muestra con el nombre DigitalPetStore.
2. Anote el ID del almacén de políticas del nuevo almacén de políticas.

3. IsAuthorizedEn la referencia de la API de permisos verificados, copia el cuerpo de la solicitud del ejemplo 4 que hace referencia al DigitalPetStoreejemplo.
4. Abre tu cliente de API y crea una solicitud al punto final del servicio regional de tu almacén de políticas. Rellene los encabezados como se muestra en el ejemplo.
5. Pegue el ejemplo del cuerpo de la solicitud y cambie el valor por el ID del almacén de `policyStoreId` políticas que indicó anteriormente.
6. Envía la solicitud y revisa los resultados. En un almacén DigitalPetStorede políticas predeterminado, esta solicitud devuelve una `ALLOW` decisión.

Puede realizar cambios en las políticas, el esquema y las solicitudes de su entorno de prueba para cambiar los resultados y tomar decisiones más complejas.

1. Cambia la solicitud de forma que modifique la decisión de Verified Permissions. Por ejemplo, cambia el rol de Alice a `Employee` o cambia el `owner` atributo del pedido 1234 a`Bob`.
2. Cambie las políticas de manera que afecten a las decisiones de autorización. Por ejemplo, modifique la política con la descripción Customer Role: Get Order para eliminar la condición de que `User` debe ser el propietario del pedido `Resource` y modifique la solicitud para que `Bob` desee ver el pedido.
3. Cambie el esquema para permitir que las políticas tomen una decisión más compleja. Actualice las entidades solicitadas para que Alice pueda cumplir con los nuevos requisitos. Por ejemplo, edite el esquema `User` para poder ser miembro de `ActiveUsers` o`InactiveUsers`. Actualice la política para que solo los usuarios activos puedan ver sus propios pedidos. Actualice las entidades de la solicitud para que Alice sea una usuaria activa o inactiva.

Integración de sus modelos de autorización con las aplicaciones

Para implementar los permisos verificados de Amazon en tu aplicación, debes definir las políticas y el esquema que deseas que aplique tu aplicación. Una vez establecido y probado el modelo de autorización, el siguiente paso es empezar a generar solicitudes de API desde el punto de aplicación. Para ello, debe configurar la lógica de la aplicación para recopilar los datos de los usuarios y rellenarlos para las solicitudes de autorización.

Cómo autoriza una aplicación las solicitudes con permisos verificados

1. Recopila información sobre el usuario actual. Por lo general, los detalles de un usuario se proporcionan en los detalles de una sesión autenticada, como un JWT o una cookie de sesión

web. Estos datos de usuario pueden provenir de una fuente de Amazon Cognito identidad vinculada a su almacén de políticas o de otro proveedor de OpenID Connect (OIDC).

2. Recopile información sobre el recurso al que un usuario quiere acceder. Por lo general, la aplicación recibirá información sobre el recurso cuando un usuario haga una selección que requiera que la aplicación cargue un nuevo activo.
3. Determina la acción que el usuario quiere realizar.
4. Genera una solicitud de autorización para Verified Permissions con el principal, la acción, el recurso y las entidades que el usuario intentó realizar la operación. Verified Permissions evalúa la solicitud comparándola con las políticas de tu almacén de políticas y devuelve una decisión de autorización.
5. La aplicación lee la respuesta de autorización o denegación de Verified Permissions y aplica la decisión sobre la solicitud del usuario.

Las operaciones de la API de permisos verificados están integradas AWS SDKs. Para incluir los permisos verificados en una aplicación, integra el AWS SDK del idioma que elijas en el paquete de la aplicación.

Para obtener más información y descargarlo AWS SDKs, consulta Herramientas para Amazon Web Services.

Los siguientes son enlaces a la documentación de varios recursos sobre permisos verificados AWS SDKs.

- AWS SDK para .NET
- AWS SDK para C++
- AWS SDK para Go
- AWS SDK para Java
- AWS SDK para JavaScript
- AWS SDK para PHP
- AWS SDK para Python (Boto)
- AWS SDK para Ruby
- AWS SDK para Rust

El siguiente AWS SDK para JavaScript ejemplo `IsAuthorized` se origina en Simplifique la autorización detallada con Amazon Verified Permissions y Amazon Cognito.

```
const authResult = await avp.isAuthorized({
    principal: 'User::"alice"',
    action: 'Action::"view"',
    resource: 'Photo::"VacationPhoto94.jpg"',
    // whenever our policy references attributes of the entity,
    // isAuthorized needs an entity argument that provides
    // those attributes
    entities: {
       entityList: [
         {
            "identifier": {
                "entityType": "User",
                "entityId": "alice"
            },
            "attributes": {
                "location": {
                    "String": "USA"
                }
            }
         }
       ]
    }
});
```

Más recursos para desarrolladores

- Taller sobre permisos verificados de Amazon
- Permisos verificados de Amazon - Recursos
- Implemente un proveedor de políticas de autorización personalizado para aplicaciones de ASP.NET Core mediante Amazon Verified Permissions
- Cree un servicio de asignación de derechos para aplicaciones empresariales con Amazon Verified Permissions
- Simplifique la autorización detallada con Amazon Verified Permissions y Amazon Cognito

Cómo realizar solicitudes a la API

Las solicitudes de consulta para los permisos verificados de Amazon son solicitudes HTTP o HTTPS que utilizan el `POST` método.

Puntos finales de permisos verificados

Un punto final es una URL que sirve como punto de entrada para un servicio web. Puede seleccionar un Región de AWS punto final adecuado al realizar sus solicitudes para reducir la latencia. Para obtener información sobre los puntos de enlace que utilizan los permisos verificados, consulte Amazon Verified Permissions en. Referencia general de Amazon Web Services

Parámetros de solicitud

Amazon Verified Permissions utiliza un JSON-based protocolo. Los parámetros de la solicitud se transmiten como JSON en el cuerpo de la solicitud HTTP. Las listas se representan como matrices JSON. Para obtener más información, consulta Parámetros comunes en la referencia de la API de permisos verificados de Amazon.

Identificadores de solicitud

En cada respuesta, AWS devuelve un ID de solicitud a través del encabezado de respuesta `x-amzn-RequestId` HTTP. Esta cadena es un identificador único que se AWS asigna para proporcionar información de seguimiento. Si bien el identificador de solicitud se incluye en todas las respuestas, no aparece en las páginas individuales de documentación de la API para mejorar la legibilidad y reducir la redundancia.

Consulta la autenticación de la API

Las solicitudes de consulta se envían a través de HTTPS. Debes incluir una firma en todas las solicitudes de consulta. Para obtener más información sobre cómo crear e incluir una firma, consulte Firmar solicitudes de AWS API en Referencia general de Amazon Web Services.

Bibliotecas disponibles

AWS proporciona bibliotecas, códigos de muestra, tutoriales y otros recursos para los desarrolladores de software que prefieren crear aplicaciones mediante API específicas del idioma en

lugar de las herramientas de línea de comandos y la API de consulta. Estas bibliotecas proporcionan funciones básicas (no incluidas en las API), como la autenticación de solicitudes, los reintentos de solicitudes y la gestión de errores, para que sea más fácil empezar. Las bibliotecas y los recursos de permisos verificados están disponibles para los siguientes lenguajes y plataformas:

- AWS SDK para Go
- AWS SDK para Java 2.x
- AWS SDK para Java 1.x
- AWS SDK para JavaScript
- AWS SDK para .NET
- AWS SDK para PHP
- AWS SDK para Python (Boto)
- AWS SDK para Ruby

Para obtener más información sobre las bibliotecas y los códigos de muestra en todos los idiomas, consulte Ejemplos de códigos y bibliotecas.

Realizar solicitudes a la API mediante el método **POST**

Si no utilizas uno de los AWS SDK, puedes realizar solicitudes de permisos verificados a través de HTTPS mediante el método de POST solicitud. El POST método requiere que especifiques la operación en el encabezado de la solicitud y proporciones los datos de la operación en formato JSON en el cuerpo de la solicitud.

Nombre del encabezado	Valor del encabezado
`Host`	El punto final de permisos verificados de Amazon. Por ejemplo: `verifiedpermissions.us-east-1.amazonaws.com`
`X-Amz-Date`	Debe proporcionar la marca de tiempo en el encabezado HTTP Date o en el encabezado AWS x-amz-date. Algunas bibliotecas cliente HTTP no permiten configurar el encabezado Date. Cuando hay un encabezado x-amz-date, el sistema ignora cualquier encabezado de fecha durante la autenticación de la solicitud.

Nombre del encabezado	Valor del encabezado
	El encabezado x-amz-date debe especificarse en el formato básico ISO 8601. Por ejemplo: `20130315T092054Z`
`Authorization`	El conjunto de parámetros de autorización que se AWS utiliza para garantizar la validez y autenticidad de la solicitud. Para obtener más información sobre la creación de este encabezado, consulte el proceso de firma de la versión 4 en el Referencia general de Amazon Web Services.
`X-Amz-Target`	Especifica la operación de permisos verificados que desea realizar. `VerifiedPermissions.` *`API_Name`* Por ejemplo, para llamar a la `CreatePolicy` operación, utilice el siguiente valor objetivo. `VerifiedPermissions.CreatePolicy`
`Content-Type`	Especifica el formato de entrada. Utilice el siguiente valor. `application/x-amz-json-1.0`
`Accept`	Especifica el formato de respuesta. Utilice el siguiente valor. `application/x-amz-json-1.0`
`Content-Length`	Tamaño de la carga útil en bytes.
`Content-Encoding`	Especifica el formato de codificación de la entrada y la salida. Utilice el siguiente valor. `amz-1.0`

A continuación se muestra un ejemplo de encabezado para que una solicitud HTTP devuelva una lista de todas las políticas del almacén de políticas especificado en el Cuenta de AWS que `Principal` hacen referencia a un `User` nombre`alice`. En este ejemplo, la `Authorization` línea está redactada aquí para facilitar la lectura. No lo pongas en palabras en tu solicitud real.

```
POST / HTTP/1.1
Host: verifiedpermissions.us-east-1.amazonaws.com
X-Amz-Date: 20230101T200059Z
Accept-Encoding: identity
Content-Type: application/x-amz-json-1.0
X-Amz-Target: VerifiedPermissions.ListPolicies
User-Agent: <UserAgentString>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>, SignedHeaders=<Headers>,
 Signature=<Signature>
Content-Length: <PayloadSizeBytes>

{
    "PolicyStoreId": "PSExAmPLE222222222",
    "Filter": {
        "Principal": {
            "Id": {
                "EntityType": "User",
                "EntityId": "alice"
            }
        }
    }
}
```

Seguridad en Amazon Verified Permissions

La seguridad en la nube AWS es la máxima prioridad. Como AWS cliente, usted se beneficia de los centros de datos y las arquitecturas de red diseñados para cumplir con los requisitos de las organizaciones más sensibles a la seguridad.

La seguridad es una responsabilidad compartida entre AWS usted y usted. El modelo de responsabilidad compartida la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta AWS los servicios en la Nube de AWS. AWS también le proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los AWS programas de de . Para obtener más información sobre los programas de conformidad que se aplican a los permisos verificados de Amazon, consulte AWS Servicios incluidos en el ámbito del programa de conformidad AWS .
- Seguridad en la nube: su responsabilidad viene determinada por el AWS servicio que utilice. También eres responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y la normativa aplicables.

Esta documentación le ayuda a comprender cómo aplicar el modelo de responsabilidad compartida cuando se utiliza Verified Permissions. En los siguientes temas, se le mostrará cómo configurar Verified Permissions para satisfacer sus objetivos de seguridad y conformidad. También aprenderá a utilizar otros AWS servicios que le ayudan a supervisar y proteger sus recursos de permisos verificados.

Temas

- Protección de los datos en Amazon Verified Permissions
- Administración de identidades y accesos de Amazon Verified Permissions
- Validación de conformidad de Amazon Verified Permissions
- Resiliencia de Amazon Verified Permissions

Protección de los datos en Amazon Verified Permissions

El modelo de se aplica a protección de datos en Amazon Verified Permissions. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta toda la Nube

de AWS. Eres responsable de mantener el control sobre el contenido alojado en esta infraestructura. Este contenido incluye las tareas de configuración y administración de la seguridad Servicios de AWS que utilizas. Para obtener más información sobre la privacidad de los datos, consulte las Preguntas frecuentes sobre la privacidad de datos. Para obtener información sobre la protección de datos en Europa, consulte la publicación de blog sobre el Modelo de responsabilidad compartida de AWS y GDPR en el Blog de seguridad de AWS .

- Con fines de protección de datos, le recomendamos que proteja Cuenta de AWS las credenciales y configure los usuarios individuales con AWS IAM Identity Center o AWS Identity and Access Management (IAM). De esta manera, solo se otorgan a cada usuario los permisos necesarios para cumplir sus obligaciones laborales.
- Recomendamos que proteja sus datos de las siguientes formas:
  - Utiliza la autenticación multifactor (MFA) en cada cuenta.
  - Se utiliza SSL/TLS para comunicarse con AWS los recursos. Se requiere usar TLS 1.2.
  - Configure la API y el registro de actividad de los usuarios con AWS CloudTrail.
  - Utilice soluciones de AWS cifrado, junto con todos los controles de seguridad predeterminados Servicios de AWS.
  - Utilice servicios de seguridad gestionados avanzados Amazon Macie, como los que ayudan a descubrir y proteger los datos confidenciales almacenados en ellos Amazon S3.
  - Si necesita módulos criptográficos validados FIPS 140-2 al acceder a AWS a través de una interfaz de línea de comandos o una API, utilice un punto de conexión de FIPS. Para obtener más información sobre los puntos de conexión de FIPS disponibles, consulte Estándar de procesamiento de la información federal (FIPS) 140-2.
- Se recomienda encarecidamente no introducir nunca información confidencial o sensible, como, por ejemplo, direcciones de correo electrónico de clientes, en etiquetas o campos de formato libre, tales como el campo Nombre. Esto incluye cuando trabaja con permisos verificados u otros tipos de permisos Servicios de AWS mediante la consola AWS CLI, la API o AWS SDKs. Cualquier dato que introduzca en etiquetas o campos de formato libre utilizados para los nombres se pueden emplear para los registros de facturación o diagnóstico. Si proporciona una URL a un servidor externo, recomendamos encarecidamente que no incluya información de credenciales en la URL a fin de validar la solicitud para ese servidor.
- Los nombres de sus acciones no deben incluir información confidencial.
- También le recomendamos encarecidamente que utilice siempre identificadores únicos, no mutables ni reutilizables para sus entidades (recursos y entidades principales). En un entorno de prueba, puede optar por utilizar identificadores de entidad simples, como `jane` o `bob` para el

nombre de una entidad de tipo `User`. Sin embargo, en un sistema de producción, es fundamental, por motivos de seguridad, utilizar valores únicos que no se puedan reutilizar. Te recomendamos que utilices valores como los identificadores únicos universales (UUIDs). Por ejemplo, piense en el usuario `jane` que deja la empresa. Más adelante, deja que otra persona use el nombre `jane`. Ese nuevo usuario tiene acceso automáticamente a todo lo que otorgan las políticas que aún hacen referencia a `User::"jane"`. Verified Permissions y Cedar no pueden distinguir entre el usuario nuevo y el anterior.

Esta directriz se aplica a los identificadores de las entidades principales y de los recursos. Utilice siempre identificadores con garantías de que son únicos y que no se han reutilizado nunca para asegurarse de no conceder acceso involuntariamente debido a la presencia de un identificador antiguo en una política.

- Asegúrese de que las cadenas que proporciona para definir los valores `Long` y `Decimal` estén dentro del rango válido de cada tipo. Además, asegúrese de que el uso de cualquier operador aritmético no dé como resultado un valor fuera del rango válido. Si se supera el rango, la operación produce una excepción de desbordamiento. Se omite una política que dé lugar a un error, lo que significa que una política de permisos podría no permitir el acceso de forma inesperada o que una política de prohibición podría no bloquear el acceso de forma inesperada.

Cifrado de datos

Amazon Verified Permissions cifra automáticamente todos los datos de los clientes, como las políticas, con un Clave administrada de AWS. Los permisos verificados de Amazon también permiten a los clientes utilizar un clave administrada por el cliente para cifrar sus datos.

Para obtener información detallada sobre el uso de claves administradas por el cliente para el cifrado, consultethe section called “Claves administradas por el cliente”.

Cifrado de recursos en permisos verificados de Amazon

Amazon Verified Permissions proporciona cifrado de forma predeterminada para proteger los datos confidenciales de los clientes en reposo mediante claves de cifrado AWS propias. Como capa adicional de protección, Amazon Verified Permissions te permite cifrar tus almacenes de pólizas mediante AWS Key Management Service (AWS KMS) clave administrada por el cliente s (CMK). Esta funcionalidad garantiza la protección de los datos confidenciales mediante el cifrado en reposo, lo que le ayuda a:

- Reduzca la carga operativa de su aplicación para proteger los datos confidenciales
- Mantenga el control sobre quién puede ver los detalles de sus políticas de autorización a través de AWS KMS clave administrada por el cliente los suyos
- Creación de aplicaciones sensibles a la seguridad que necesitan cumplimiento estricto de cifrado y requisitos normativos

En las siguientes secciones se explica cómo configurar el cifrado para los nuevos almacenes de políticas y cómo administrar las claves de cifrado.

AWS KMS Tipos de claves para los permisos verificados de Amazon

Amazon Verified Permissions se integra AWS KMS para gestionar las claves de cifrado utilizadas para los datos de los encrypting/decrypting clientes. Para obtener más información acerca de los tipos y estados de las claves, consulte los conceptos de AWS Key Management Service en la Guía para desarrolladores de AWS KMS . Al crear un nuevo almacén de políticas, puede elegir entre los siguientes tipos de AWS KMS claves para cifrar sus datos:

AWS Clave propia

El tipo de cifrado predeterminado. Amazon Verified Permissions es el propietario de la clave sin coste adicional para usted y cifra los datos de los recursos en reposo en el momento de su creación. No es necesario realizar ninguna configuración adicional en el código o las aplicaciones de encrypt/decrypt sus datos mediante la clave propiedad de Verified Permissions.

Clave gestionada por el cliente

Tú creas, eres el propietario y administras la clave de tu AWS cuenta. Tienes el control total sobre la AWS KMS clave. AWS KMS se aplican cargos para clave administrada por el cliente nosotros. Para obtener más información, consulte la Página de precios de AWS KMS . Para obtener más información sobre los tipos de claves, consulte las claves administradas por el cliente en la Guía para AWS KMS desarrolladores.

Al especificar una clave administrada por el cliente para el cifrado de los recursos de nivel superior (es decir, el almacén de políticas), Verified Permissions cifra el recurso, así como sus recursos secundarios, con esa clave. Para cifrar un almacén de políticas mediante una clave administrada por el cliente, debes conceder acceso a los permisos verificados de tu política de claves. Una política de claves es una política basada en recursos que se adjunta para controlar el acceso clave administrada por el cliente a la misma. Consulte the section called “Autorizar el uso de tu AWS KMS clave para los permisos verificados de Amazon” para obtener más detalles.

Además, para crear un almacén de políticas cifrado con un almacén de políticas cifrado por un clave administrada por el cliente, o para realizar llamadas a la API a un almacén de políticas cifrado por un clave administrada por el cliente, el usuario o rol de IAM que realiza la llamada también debe tener acceso a la clave. Si Verified Permissions no puede acceder a la clave, cualquier decisión de autorización que involucre recursos cifrados por esa clave puede estar obsoleta o ser inexacta. Si no tiene acceso a la clave, no podrá read/update /eliminar los recursos cifrados por esa clave, y cualquier llamada creada para utilizarla con fines de cifrado fallará.

> **Note**
>
> El cifrado de permisos verificados en reposo está disponible en todas AWS las regiones en las que están disponibles los permisos verificados.

> **Important**
>
> Una vez que se clave administrada por el cliente ha utilizado para cifrar un almacén de políticas, NO PUEDE actualizar el recurso para utilizar una clave de cifrado diferente ni eliminar la clave de ese almacén de políticas.

Utilización AWS KMS y claves de datos con permisos verificados de Amazon

La función de cifrado en reposo de Amazon Verified Permissions utiliza una AWS KMS clave y una jerarquía de claves de datos para proteger los datos de sus recursos.

> **Note**
>
> Amazon Verified Permissions solo admite AWS KMS claves simétricas. No puedes usar una AWS KMS clave asimétrica para cifrar tus recursos de permisos verificados de Amazon.

Utilización AWS Claves propias

Amazon Verified Permissions cifra todos los recursos de forma predeterminada con claves AWS propias. Estas claves son de uso gratuito y se rotan anualmente para proteger los recursos de la cuenta. No necesita ver, administrar, usar ni auditar estas claves, por lo que no es necesario realizar ninguna acción para proteger los datos. Para obtener más información sobre las claves AWS propias, consulta las claves AWS propias en la Guía para AWS KMS desarrolladores.

Uso de claves administradas por el cliente

La selección de una clave administrada por el cliente para el cifrado ofrece las siguientes ventajas:

- Usted crea y administra la AWS KMS clave, incluida la configuración de las políticas clave y IAM las políticas para controlar el acceso a la AWS KMS clave. Puede activar y desactivar la AWS KMS clave, activar y desactivar la rotación automática de claves y eliminar la AWS KMS clave cuando ya no esté en uso.
- Puede utilizar una clave administrada por el cliente con material clave importado o clave administrada por el cliente en un almacén de claves personalizado que sea de su propiedad y que administre.
- Puede auditar el cifrado y el descifrado de sus recursos de permisos verificados examinando las llamadas a la API de permisos verificados de Amazon AWS KMS en los AWS CloudTrail registros.

Para que Amazon Verified Permissions utilice tu clave administrada por el cliente s for encryption/decryption, tendrás que añadir políticas clave específicas para permitir que Amazon Verified Permissions acceda a encrypt/decrypt los recursos en tu nombre.

Autorizar el uso de tu AWS KMS clave para los permisos verificados de Amazon

Como mínimo, Amazon Verified Permissions requiere los siguientes permisos en un clave administrada por el cliente:

- `kms:Encrypt`
- `kms:GenerateDataKeyWithoutPlaintext`
- `kms:DescribeKey`
- `kms:ReEncryptTo`
- `kms:ReEncryptFrom`
- `kms:Decrypt`

A continuación se muestra un ejemplo de política clave:

```
{
    "Sid": "Enable AVP to use the KMS key for encrypting project J.A.K. policy
 resources",
    "Effect": "Allow",
```

```
    "Principal": {
        "Service": "verifiedpermissions.amazonaws.com"
    },
    "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Encrypt",
        "kms:ReEncryptFrom",
        "kms:ReEncryptTo",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}
```

Comprensión del contexto de la fuente

El contexto de la fuente proporciona información sobre la persona que llama de origen que intenta realizar AWS KMS acciones contra una clave determinada. Esto evita la confusión o el uso indebido de los datos cifrados al vincular el contexto a la fuente de los datos.

Los clientes pueden utilizar el contexto de origen como condiciones adicionales en su política clave, como las siguientes declaraciones de política clave:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Enable this account full access to this key",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Enable AVP to retrieve this key's metadata",
            "Effect": "Allow",
            "Principal": {
                "Service": "verifiedpermissions.amazonaws.com"
            },
            "Action": "kms:DescribeKey",
            "Resource": "*",
```

```
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "111122223333"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:verifiedpermissions::111122223333:policy-
store/*"
                }
            }
        },
        {
            "Sid": "Enable AVP to encrypt/decrypt resources utilizing this key",
            "Effect": "Allow",
            "Principal": {
                "Service": "verifiedpermissions.amazonaws.com"
            },
            "Action": [
                "kms:Decrypt",
                "kms:ReEncryptTo",
                "kms:ReEncryptFrom",
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:Encrypt"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "111122223333"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:verifiedpermissions::111122223333:policy-
store/*"
                }
            }
        }
    ]
}
```

Esta política clave permite a Verified Permissions realizar AWS KMS llamadas en tu nombre, si la cuenta de origen es la misma que la cuenta en la que reside la AWS KMS clave. Estos valores deben poder verificarse al comprobar los registros de AWS CloudTrail auditoría para detectar la clave CMK. Para obtener más información sobre las claves de AWS condición globales, consulte Uso `aws:SourceArn` de nuestras claves de `aws:SourceAccount` condición.

Comprensión del contexto de cifrado

El contexto de cifrado es un conjunto de pares clave-valor que contienen datos autenticados adicionales para comprobar la integridad del cifrado. Al incluir un contexto de cifrado en una solicitud de cifrado de datos, vincula AWS KMS criptográficamente el contexto de cifrado a los datos cifrados. Para descifrar los datos, debe pasar por el mismo contexto de cifrado.

Amazon Verified Permissions utiliza el mismo contexto de cifrado en todas las operaciones AWS KMS criptográficas y se pueden verificar en AWS CloudTrail los registros cuando Verified Permissions realiza AWS KMS llamadas en su nombre para encryption/decryption procesar. De forma predeterminada, Verified Permissions utiliza los siguientes pares clave-valor del contexto de cifrado al cifrar sus recursos:

```
{
    "aws:verifiedpermissions:policy-store-arn":
 "arn:aws:verifiedpermissions::111122223333:policy-store/PSt123456789012"
}
```

Los permisos verificados de Amazon también le permiten añadir un contexto de cifrado personalizado como parte de los metadatos adicionales que desee incluir durante encryption/decryption los procesos. Esto significa que su política de claves puede ser más precisa a la hora de conceder permisos, como en el ejemplo siguiente:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Enable this account full access to this key",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Enable AVP to retrieve this key's metadata",
            "Effect": "Allow",
            "Principal": {
                "Service": "verifiedpermissions.amazonaws.com"
            },
```

```
            "Action": "kms:DescribeKey",
            "Resource": "*"
        },
        {
            "Sid": "Enable AVP to encrypt/decrypt resources utilizing this key",
            "Effect": "Allow",
            "Principal": {
                "Service": "verifiedpermissions.amazonaws.com"
            },
            "Action": [
                "kms:Decrypt",
                "kms:ReEncryptTo",
                "kms:ReEncryptFrom",
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:Encrypt"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "kms:EncryptionContext:aws:verifiedpermissions:policy-store-arn":
 "arn:aws:verifiedpermissions::111122223333:policy-store/*",
                    "kms:EncryptionContext:policy_owner": "Tim"
                }
            }
        }
    ]
}
```

Esta política de claves permite a Verified Permissions realizar AWS KMS llamadas en tu nombre si el mapa de contexto de cifrado contiene una clave `aws:verifiedpermissions:policy-store-arn` cuyo valor sigue el formato de `arn:aws:verifiedpermissions::111122223333:policy-store/*` y también contiene un par clave-valor. `"policy_owner": "Tim"` Consulte the section called “Creación de un almacén de políticas cifradas” para saber cómo configurar un contexto de cifrado personalizado.

> **Note**
>
> Se recomienda que las políticas clave con condiciones basadas en el contexto de cifrado pertenezcan a un subconjunto del mapa de contexto de cifrado, en lugar de comprobar cada par clave-valor. Es posible que el servicio y sus dependencias iniciales agreguen pares clave-valor adicionales que no estén visibles para usted y, si la política de claves lo permite

condicionalmente, pueden afectar al acceso a las claves de Verified Permissions, en función del aspecto exacto del mapa contextual de cifrado.

Entendiendo los kilómetros: ViaService

La clave de `kms:ViaService` condición limita el uso de una AWS KMS clave a las solicitudes de AWS servicios específicos. Esta clave de condición solo se aplica a las sesiones de acceso directo (FAS). Para obtener más información`kms:ViaService`, consulte kms: ViaService en la Guía para AWS KMS desarrolladores.

Por ejemplo, la siguiente declaración de política clave utiliza la clave de `kms:ViaService` condición para permitir que clave administrada por el clientea se utilice para las acciones especificadas solo cuando la solicitud proviene de Amazon Verified Permissions en la región EE.UU. Este (Virginia del Norte) en nombre de`BrentRole`.

```
{
    "Sid": "Enable AVP to encrypt/decrypt resources using credentials of BrentRole",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/BrentRole"
    },
    "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Encrypt",
        "kms:ReEncryptFrom",
        "kms:ReEncryptTo",
        "kms:DescribeKey"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "kms:ViaService": [
                "verifiedpermissions.us-east-1.amazonaws.com"
            ]
        }
    }
}
```

Esto es necesario para que Verified Permissions pueda transmitir su identidad, permisos y atributos de sesión cuando Verified Permissions lo solicite AWS KMS en su nombre encryption/decryption. Para obtener más información sobre las solicitudes de FAS, consulte las sesiones de acceso directo en la Guía del IAM usuario.

Completado AWS KMS Política clave

Basado en los conceptos de las secciones anteriores, este es un ejemplo de política clave que permitirá a Amazon Verified Permissions utilizar una CMK para encryption/decryption:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Enable this account full access to this key",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Enable AVP to retrieve this key's metadata",
            "Effect": "Allow",
            "Principal": {
                "Service": "verifiedpermissions.amazonaws.com"
            },
            "Action": "kms:DescribeKey",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "111122223333"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:verifiedpermissions::111122223333:policy-
store/*"
                }
            }
        },
        {
            "Sid": "Enable AVP to encrypt/decrypt resources utilizing this key",
            "Effect": "Allow",
```

```
            "Principal": {
                "Service": "verifiedpermissions.amazonaws.com"
            },
            "Action": [
                "kms:Decrypt",
                "kms:ReEncryptTo",
                "kms:ReEncryptFrom",
                "kms:Encrypt",
                "kms:GenerateDataKeyWithoutPlaintext"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "kms:EncryptionContext:aws:verifiedpermissions:policy-store-arn":
 "arn:aws:verifiedpermissions::111122223333:policy-store/*",
                    "kms:EncryptionContext:policy_owner": "Tim"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:verifiedpermissions::111122223333:policy-
store/*"
                },
                "StringEquals": {
                    "aws:SourceAccount": "111122223333"
                }
            }
        },
        {
            "Sid": "Enable AVP to encrypt/decrypt resources using credentials of
 BrentRole",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/BrentRole"
            },
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:Encrypt",
                "kms:ReEncryptFrom",
                "kms:ReEncryptTo"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": [
```

```
                        "verifiedpermissions.us-east-1.amazonaws.com"
                    ]
                },
                "StringLike": {
                    "kms:EncryptionContext:aws:verifiedpermissions:policy-store-arn":
 "arn:aws:verifiedpermissions::111122223333:policy-store/*",
                    "kms:EncryptionContext:policy_owner": "Tim"
                }
            }
        },
        {
            "Sid": "Enable AVP to describe this key using credentials of BrentRole",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/BrentRole"
            },
            "Action": "kms:DescribeKey",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": [
                        "verifiedpermissions.us-east-1.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

Warning

Tenga cuidado al modificar las políticas AWS KMS clave para las claves que Amazon Verified Permissions ya utiliza. Si bien Verified Permissions valida los permisos de cifrado y descifrado cuando se configura inicialmente una AWS KMS clave durante la creación de un recurso de nivel superior, no puede verificar los cambios de política posteriores a pedido. La eliminación inadvertida de los permisos necesarios podría interrumpir sus decisiones de autorización y los flujos habituales del servicio de permisos verificados. Para obtener orientación sobre la solución de errores comunes relacionados clave administrada por el cliente con los permisos verificados de Amazon, consultethe section called “Solucionar problemas con las claves gestionadas por el cliente en los permisos verificados de Amazon”.

Necesario IAM Políticas de recursos cifrados

Los clientes que llamen a Verified Permissions a través de un IAM rol dentro de su cuenta deberán asegurarse de que la IAM política correspondiente cuente con los permisos adecuados clave administrada por el cliente para utilizarlos para cifrar y descifrar los recursos.

Para crear almacenes de políticas cifrados mediante un clave administrada por el cliente, la siguiente IAM política ilustra las acciones mínimas necesarias AWS KMS y las relacionadas con los permisos verificados para hacerlo:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "verifiedpermissions:CreatePolicyStore",
            "Resource": "*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "kms:Decrypt",
                "kms:Encrypt",
                "kms:ReEncryptTo",
                "kms:ReEncryptFrom",
                "kms:DescribeKey",
                "kms:GenerateDataKeyWithoutPlaintext"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

> **Note**
>
> Para recuperar (operaciones Get* y List*) y eliminar los almacenes de políticas cifrados por a clave administrada por el cliente, no se necesitan permisos adicionales.

Para actualizar un almacén de políticas cifrado por a clave administrada por el cliente, recuperar (operaciones Get* y List*), actualizar y eliminar los recursos secundarios de un almacén de políticas cifrado por a clave administrada por el cliente, la siguiente IAM política ilustra las acciones mínimas necesarias AWS KMS y verificadas para ello:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "verifiedpermissions:*",
            "Resource": "*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

Como IAM política única, los clientes simplemente pueden añadir lo siguiente a su política de funciones: IAM

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": "verifiedpermissions:*",
            "Resource": "*",
            "Effect": "Allow"
        },
        {
            "Action": [
                "kms:Decrypt",
                "kms:Encrypt",
                "kms:ReEncryptTo",
```

```
                "kms:ReEncryptFrom",
                "kms:DescribeKey",
                "kms:GenerateDataKeyWithoutPlaintext"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

Administración de almacenes de políticas cifradas

Los almacenes de políticas son el contenedor básico que contendrá todos los recursos de políticas relacionados. Para obtener más información sobre los almacenes de políticas y la jerarquía de los recursos secundarios, consulte los almacenes de políticas de permisos verificados de Amazon en la Guía del usuario de permisos verificados de Amazon.

Al crear un almacén de políticas en Verified Permissions, puedes activar el cifrado en reposo mediante AWS KMS claves. Esto garantiza que:

- Todas las operaciones de lectura, actualización y eliminación de los almacenes de políticas y sus recursos secundarios utilizarán lo proporcionado clave administrada por el cliente para los procesos de descifrado
- Cualquier llamada de decisión de autorización (es decir IsAuthorized BatchIsAuthorized, IsAuthorizedWithToken,, etc.) utilizará lo proporcionado clave administrada por el cliente para los procesos de descifrado

Creación de un almacén de políticas cifradas

Antes de crear un almacén de políticas cifrado, asegúrate de que el que clave administrada por el cliente estás utilizando tenga establecidas las declaraciones de políticas clave adecuadas para que Amazon Verified Permissions utilice la clave encryption/decryption. Consulta the section called “Autorizar el uso de tu AWS KMS clave para los permisos verificados de Amazon” qué permisos son necesarios.

Uso de AWS CLI:

```
aws verifiedpermissions create-policy-store --region us-east-1 --encryption-settings
 file://encrypted.json --validation-settings "{\"mode\": \"OFF\"}"
```

Dónde `encrypted.json` se ve así:

```
{
    "kmsEncryptionSettings": {
        "key": "arn:aws:kms:us-east-1:111122223333:key/12345678-90ab-cdef-ghij-klmnopqrstuv",
        "encryptionContext": {
            "<ENCRYPTION_CONTEXT_KEY_1>": "<ENCRYPTION_CONTEXT_VALUE_1>",
            "<ENCRYPTION_CONTEXT_KEY_2>": "<ENCRYPTION_CONTEXT_VALUE_2>",
            ...
        }
    }
}
```

Asegúrese de reemplazar por `key` su clave administrada por el cliente ARN y de reemplazar `<ENCRYPTION_CONTEXT_KEY>` y por los `<ENCRYPTION_CONTEXT_VALUE>` pares clave-valor deseados`encryptionContext`. `encryptionContext`se puede omitir por completo si no se desea sumar pares clave-valor.

> **Important**
>
> No incluya el par clave-valor `aws:verifiedpermissions:policy-store-arn` en su contexto de cifrado personalizado. Esto se agrega automáticamente y provocará errores de validación si forma parte de los pares clave-valor del contexto de cifrado personalizado que haya aprobado.

Para obtener más información sobre las API disponibles de los recursos secundarios de un almacén de políticas, consulte Acciones en la Guía de referencia de la API de permisos verificados de Amazon.

> **Note**
>
> Si los AWS KMS clave administrada por el cliente recursos de permisos verificados de Amazon que están utilizando se eliminan, deshabilitan o no se puede acceder a ellos debido a una política de AWS KMS claves incorrecta, el descifrado de los recursos fallará y, por lo tanto, las decisiones de autorización quedarán obsoletas. La pérdida de acceso puede ser temporal (se puede corregir una política de claves) o permanente (no se puede restaurar una clave eliminada), según las circunstancias. Le recomendamos que restrinja el acceso

a las operaciones críticas, como eliminar o deshabilitar la clave. AWS KMS Además, le recomendamos que su organización establezca procedimientos de acceso AWS innovadores para garantizar que sus usuarios privilegiados puedan acceder AWS en el improbable caso de que Amazon Verified Permissions no esté accesible.

Supervisión de la interacción de permisos verificados de Amazon con AWS KMS

Puedes supervisar el uso que Amazon Verified Permissions hace de tu clave administrada por el cliente canal. AWS CloudTrail Cada solicitud a AWS KMS través de permisos verificados incluye el contexto de cifrado y la clave ARN que se utiliza (su clave administrada por el cliente) en los parámetros de la solicitud:

Ejemplo de entrada de AWS CloudTrail registro para`GenerateDataKeyWithoutPlaintext`:

```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "verifiedpermissions.amazonaws.com"
    },
    "eventTime": "2025-09-28T16:51:04Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKeyWithoutPlaintext",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "verifiedpermissions.amazonaws.com",
    "userAgent": "verifiedpermissions.amazonaws.com",
    "requestParameters": {
        "keyId": "arn:aws:kms:us-east-1:111122223333:key/abcdefgh-0123-ijkl-4567-
mnopqrstuvwx",
        "encryptionContext": {
            "aws:verifiedpermissions:policy-store-arn":
 "arn:aws:verifiedpermissions::111122223333:policy-store/PSt123456789012",
            "policy_store_editor": "Janus"
        },
        ...
    },
    ...
}
```

Ejemplo de entrada de AWS CloudTrail registro para`Decrypt`:

```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "verifiedpermissions.amazonaws.com"
    },
    "eventTime": "2025-09-28T16:53:21Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "verifiedpermissions.amazonaws.com",
    "userAgent": "verifiedpermissions.amazonaws.com",
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "keyId": "arn:aws:kms:us-east-1:111122223333:key/abcdefgh-0123-ijkl-4567-
mnopqrstuvwx",
        "encryptionContext": {
            "aws:verifiedpermissions:policy-store-arn":
 "arn:aws:verifiedpermissions::111122223333:policy-store/PSt123456789012",
            "policy_store_owner": "Elias"
        }
    },
    ...
}
```

Ejemplo de entrada de AWS CloudTrail registro para`ReEncrypt`:

```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "verifiedpermissions.amazonaws.com"
    },
    "eventTime": "2025-09-28T16:51:04Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ReEncrypt",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "verifiedpermissions.amazonaws.com",
    "userAgent": "verifiedpermissions.amazonaws.com",
    "requestParameters": {
        "sourceKeyId": "arn:aws:kms:us-east-1:111122223333:key/abcdefgh-0123-ijkl-4567-
mnopqrstuvwx",
        "destinationEncryptionContext": {
```

```
            "aws:verifiedpermissions:policy-store-arn":
 "arn:aws:verifiedpermissions::111122223333:policy-store/PSt123456789012"
        },
        "sourceEncryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "destinationKeyId": "arn:aws:kms:us-east-1:111122223333:key/abcdefgh-0123-
ijkl-4567-mnopqrstuvwx",
        "sourceEncryptionContext": {
            "aws:verifiedpermissions:policy_store_arn":
 "arn:aws:verifiedpermissions::111122223333:policy-store/PSt123456789012"
        },
        "destinationEncryptionAlgorithm": "SYMMETRIC_DEFAULT",
        ...
    },
    ...
}
```

Tenga en cuenta que las entradas del registro incluyen `invokedBy` referencias al director de Amazon Verified Permissions y `encryptionContext/sourceEncryptionContext/destinationEncryptionContext` están incluidas en el mapa. `requestParameters`

Ejemplo de entrada de AWS CloudTrail registro para: `DescribeKey`

```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "verifiedpermissions.amazonaws.com"
    },
    "eventTime": "2025-09-28T16:51:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "verifiedpermissions.amazonaws.com",
    "userAgent": "verifiedpermissions.amazonaws.com",
    "requestParameters": {
        "keyId": "arn:aws:kms:us-east-1:111122223333:key/abcdefgh-0123-ijkl-4567-
mnopqrstuvwx"
    },
    ...
}
```

Tenga en cuenta que la entrada del registro incluye una `invokedBy` referencia al director de Amazon Verified Permissions.

Para obtener más información sobre las entradas de AWS CloudTrail registro, consulte Descripción de los AWS CloudTrail eventos en la Guía del AWS CloudTrail usuario.

Limitaciones

En este tema se describen las limitaciones actuales de los permisos verificados y el uso de clave administrada por el cliente s para cifrar los recursos.

- No puede deshabilitar el cifrado de un almacén de políticas una vez activado
- Después de crear un almacén de políticas sin cifrado, no puede actualizar el almacén de políticas para que lo cifre un clave administrada por el cliente
- Tras revocar los permisos verificados de acceso a un almacén clave administrada por el cliente de políticas cifradas existente, es posible que las decisiones de autorización queden obsoletas
- Después de crear un almacén de políticas con un clave administrada por el cliente, no puede modificar los valores de contexto de cifrado personalizados; son valores estáticos que se establecen durante la creación del almacén de políticas cifrado

Solucionar problemas con las claves gestionadas por el cliente en los permisos verificados de Amazon

En este tema se describen los errores clave administrada por el cliente relacionados más comunes que se pueden encontrar al utilizar los permisos verificados de Amazon y se proporcionan los pasos de solución de problemas para resolverlos.

Acceso denegado: AWS KMS Problema de permiso

Error: «El servicio o la persona que llama no están autorizados a usar la AWS KMS clave proporcionada porque el recurso no existe en esta región, no hay políticas basadas en recursos que permitan el acceso o porque una política basada en recursos deniega el acceso de forma explícita»

Esto podría significar que el servicio o la persona que llama carece de los permisos de `kms:*` acción requeridos en su IAM política/política AWS KMS clave o que la clave a la que se hace referencia no existe o ya no existe.

Solución de problemas con AWS CloudTrail:

- Busque `kms.amazonaws.com` eventos en AWS CloudTrail
- Busque el nombre del evento de la AWS KMS operación que se identificó como no permitida (es decir `DecryptReEncrypt`,`GenerateDataKeyWithoutPlaintext`,`DescribeKey`,, etc.)

- Revise los campos `errorCode` y `errorMessage`
- Compruebe `userIdentity` para confirmar qué entidad principal intentó realizar la operación

Para resolver este problema, conceda al usuario o IAM director los permisos de acceso a la AWS KMS operación adecuados en su IAM política y política AWS KMS clave. Para obtener más información, consulte the section called “Completado AWS KMS Política clave”.

Excepción de validación: AWS KMS Configuración clave

Error: « AWS KMS La clave configurada no tiene una configuración válida»

Esto significa que el servicio no puede utilizar la clave a la que se hace referencia para el clave administrada por el cliente cifrado debido a su configuración actual. Los motivos pueden incluir que la clave esté deshabilitada, que la clave no sea compatible EncryptionAlgorithm o que sea de un tipo no compatible KeyUsage .

Excepción de limitación: AWS KMS Límites de frecuencia

Error: «Has superado la frecuencia a la que puedes llamar» AWS KMS

Este error significa que ha superado el AWS KMS límite de operaciones criptográficas de su clave: https://docs.aws.amazon.com/kms/latest/developerguide/requests-per-second.html.

Información relacionada

- Administrar almacenes de políticas de permisos verificados
- AWS KMS prácticas recomendadas
- AWS KMS Contexto de cifrado
- AWS CloudTrail Integration
- AWS CloudTrail Ejemplos de entradas de registro

Administración de identidades y accesos de Amazon Verified Permissions

AWS Identity and Access Management (IAM) es una Servicio de AWS que ayuda al administrador a controlar de forma segura el acceso a AWS los recursos. IAM los administradores controlan quién puede autenticarse (iniciar sesión) y quién puede autorizarse (tener permisos) para usar los recursos de permisos verificados. IAM es uno Servicio de AWS que puede utilizar sin coste adicional.

Temas

- Público
- Autenticación con identidades
- Administración del acceso con políticas
- Cómo funciona Amazon Verified Permissions con IAM
- IAM políticas de permisos verificados
- Ejemplos de políticas basadas en identidad para Amazon Verified Permissions
- AWS políticas gestionadas para los permisos verificados de Amazon
- Solución de problemas de identidad y acceso de Amazon Verified Permissions

Público

La forma en que usas AWS Identity and Access Management (IAM) varía según tu función:

- Usuario del servicio: solicite permisos al administrador si no puede acceder a las características (consulte Solución de problemas de identidad y acceso de Amazon Verified Permissions).
- Administrador del servicio: determine el acceso de los usuarios y envíe las solicitudes de permiso (consulte Cómo funciona Amazon Verified Permissions con IAM).
- IAM administrador: escriba políticas para administrar el acceso (consulteEjemplos de políticas basadas en identidad para Amazon Verified Permissions)

Autenticación con identidades

La autenticación es la forma de iniciar sesión AWS con sus credenciales de identidad. Debe autenticarse como usuario de Usuario raíz de la cuenta de AWS IAM o asumiendo un IAM rol.

Puede iniciar sesión como una identidad federada con las credenciales de una fuente de identidad, como AWS IAM Identity Center (el Centro de identidades de IAM), la autenticación de inicio de sesión único o las credenciales. Google/Facebook Para obtener más información sobre el inicio de sesión, consulte Cómo iniciar sesión en la Cuenta de AWS en la Guía del usuario de AWS Sign-In .

Para el acceso programático, AWS proporciona un SDK y una CLI para firmar criptográficamente las solicitudes. Para obtener más información, consulte la versión 4 de AWS Signature para ver las solicitudes de API en la Guía del IAM usuario.

Cuenta de AWS usuario root

Al crear un Cuenta de AWS, se comienza con una identidad de inicio de sesión denominada usuario Cuenta de AWS raíz que tiene acceso completo a todos Servicios de AWS los recursos. Se recomiendaencarecidamente que no utilice el usuario raíz para las tareas diarias. Para las tareas que requieren credenciales de usuario raíz, consulte Tareas que requieren credenciales de usuario raíz en la Guía del IAM usuario.

Identidad federada

Como práctica recomendada, exija a los usuarios humanos que utilicen la federación con un proveedor de identidades para acceder Servicios de AWS mediante credenciales temporales.

Una identidad federada es un usuario del directorio empresarial, del proveedor de identidades web o al Directory Service que se accede Servicios de AWS mediante credenciales de una fuente de identidad. Las identidades federadas asumen roles que proporcionan credenciales temporales.

Para una administración de acceso centralizada, se recomienda AWS IAM Identity Center. Para obtener más información, consulte ¿Qué es el Centro de identidades de IAM? en la Guía del usuario de AWS IAM Identity Center .

Usuarios y grupos de IAM

Un usuario de IAM es una identidad con permisos específicos para una sola persona o aplicación. Recomendamos el uso de credenciales temporales en lugar de usuarios de IAM con credenciales de larga duración. Para obtener más información, consulte Exigir a los usuarios humanos que utilicen la federación con un proveedor de identidad para acceder AWS mediante credenciales temporales en la Guía del IAM usuario.

Un IAM grupo especifica un conjunto de usuarios de IAM y facilita la administración de los permisos para grupos grandes de usuarios. Para obtener más información, consulte Casos de uso para IAM usuarios en la Guía del IAM usuario.

IAM roles

Un IAM rol es una identidad dentro de ti Cuenta de AWS que tiene permisos específicos. Es similar a un usuario de IAM, pero no está asociado a una persona determinada. Puede asumir temporalmente un IAM rol en el Consola de administración de AWS cambiando de rol. Puedes asumir un rol llamando a una operación de AWS API AWS CLI o usando una URL personalizada. Para obtener

más información sobre los métodos de uso de los roles, consulte Uso de IAM roles en la Guía del IAM usuario.

IAM los roles con credenciales temporales son útiles en las siguientes situaciones:

- Acceso de usuario federado: para asignar permisos a una identidad federada, puede crear un rol y definir sus permisos. Cuando se autentica una identidad federada, se asocia la identidad al rol y se le conceden los permisos define el rol. Para obtener información sobre los roles para la federación, consulte Crear un rol para un proveedor de identidades externo (federación) en la Guía del IAM usuario. Si utiliza el IAM Identity Center, debe configurar un conjunto de permisos. IAM Identity Center correlaciona el conjunto de permisos con un rol en IAM para controlar a qué pueden acceder las identidades después de autenticarse. Para obtener información acerca de los conjuntos de permisos, consulta Conjuntos de permisos en la Guía del usuario de AWS IAM Identity Center .
- Permisos de usuario de IAM temporales: un usuario o rol de IAM puede asumir un IAM rol para adquirir temporalmente diferentes permisos para una tarea específica.
- Acceso entre cuentas: puede utilizar un rol de IAM para permitir que alguien (una entidad principal de confianza) de otra cuenta obtenga acceso a los recursos de su cuenta. Los roles son la forma principal de conceder acceso entre cuentas. Sin embargo, con algunos Servicios de AWS, puedes adjuntar una política directamente a un recurso (en lugar de usar un rol como proxy). Para conocer la diferencia entre las funciones y las políticas basadas en recursos para el acceso entre cuentas, consulte En qué se diferencian las IAM funciones de las políticas basadas en recursos en la Guía del usuario.IAM
- Aplicaciones en ejecución Amazon EC2: puede usar un IAM rol para administrar las credenciales temporales de las aplicaciones que se ejecutan en una instancia EC2 y que realizan solicitudes o solicitudes a la API. AWS CLI AWS Es preferible hacerlo de este modo a almacenar claves de acceso en la instancia de EC2. Para asignar un AWS rol a una instancia EC2 y ponerlo a disposición de todas sus aplicaciones, debe crear un perfil de instancia adjunto a la instancia. Un perfil de instancia contiene el rol y permite a los programas que se ejecutan en la instancia de EC2 obtener credenciales temporales. Para obtener más información, consulte Usar un IAM rol para conceder permisos a las aplicaciones que se ejecutan en las Amazon EC2 instancias en la Guía del IAM usuario.

Para saber si se deben usar IAM roles o usuarios de IAM, consulte Cuándo crear un IAM rol (en lugar de un usuario) en la Guía del IAM usuario.

Administración del acceso con políticas

El acceso se controla AWS creando políticas y adjuntándolas a AWS identidades o recursos. Una política define los permisos cuando están asociados a una identidad o un recurso. AWS evalúa estas políticas cuando un director hace una solicitud. La mayoría de las políticas se almacenan AWS como documentos JSON. Para obtener más información sobre los documentos de políticas de JSON, consulte Descripción general de las políticas de JSON en la Guía del IAM usuario.

Mediante las políticas, los administradores especifican quién tiene acceso a qué, definiendo qué entidad principal puede realizar acciones sobre qué recursos y en qué condiciones.

De forma predeterminada, los usuarios y los roles no tienen permisos. Un IAM administrador crea IAM políticas y las añade a las funciones, que luego los usuarios pueden asumir. IAM las políticas definen los permisos independientemente del método utilizado para realizar la operación.

Políticas basadas en identidades

Las políticas basadas en identidad son documentos de política de permisos JSON que asocia a una identidad (usuario, grupo o rol). Estas políticas controlan qué acciones pueden realizar las identidades, en qué recursos y en qué condiciones. Para obtener información sobre cómo crear una política basada en la identidad, consulte Definir IAM permisos personalizados con políticas administradas por el cliente en la Guía del IAM usuario.

Las políticas basadas en identidad pueden ser políticas insertadas (incrustadas directamente en una sola identidad) o políticas administradas (políticas independientes asociadas a varias identidades). Para saber cómo elegir entre políticas gestionadas e integradas, consulte Elegir entre políticas gestionadas y políticas integradas en la Guía del usuario.IAM

Políticas basadas en recursos

Las políticas basadas en recursos son documentos de políticas JSON que se asocian a un recurso. Entre los ejemplos se incluyen las políticas de confianza en los IAM roles y las políticas de Amazon S3 segmento. En los servicios que admiten políticas basadas en recursos, los administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Debe especificar una entidad principal en una política basada en recursos.

Las políticas basadas en recursos son políticas insertadas que se encuentran en ese servicio. No puedes usar políticas AWS administradas desde una política basada IAM en recursos.

Listas de control de acceso () ACLs

Las listas de control de acceso (ACLs) controlan qué responsables (miembros de la cuenta, usuarios o roles) tienen permisos para acceder a un recurso. ACLs son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de políticas JSON.

Amazon S3 AWS WAF, y Amazon VPC son ejemplos de servicios compatibles. ACLs Para obtener más información ACLs, consulte la descripción general de la lista de control de acceso (ACL) en la Guía para desarrolladores de Amazon Simple Storage Service.

Otros tipos de políticas

AWS admite tipos de políticas adicionales que pueden establecer los permisos máximos otorgados por los tipos de políticas más comunes:

- Límites de permisos: establece los permisos máximos que una política basada en la identidad puede conceder a una IAM entidad. Para obtener más información, consulte los límites de los permisos para IAM las entidades en la Guía del IAM usuario.
- Políticas de control de servicios (SCPs): especifican los permisos máximos para una organización o unidad organizativa en AWS Organizations. Para obtener más información, consulte Políticas de control de servicios en la Guía del usuario de AWS Organizations .
- Políticas de control de recursos (RCPs): establece los permisos máximos disponibles para los recursos de tus cuentas. Para obtener más información, consulte Políticas de control de recursos (RCPs) en la Guía del AWS Organizations usuario.
- Políticas de sesión: políticas avanzadas que se pasan como parámetro cuando se crea una sesión temporal para un rol o un usuario federado. Para obtener más información, consulte Políticas de sesión en la Guía del usuario de IAM .

Varios tipos de políticas

Cuando se aplican varios tipos de políticas a una solicitud, los permisos resultantes son más complicados de entender. Para saber cómo se AWS determina si se debe permitir una solicitud cuando se trata de varios tipos de políticas, consulte la lógica de evaluación de políticas en la Guía del IAM usuario.

Cómo funciona Amazon Verified Permissions con IAM

Antes de administrar el acceso IAM a los permisos verificados, obtén información sobre las IAM funciones disponibles para su uso con los permisos verificados.

IAM funciones que puedes usar con los permisos verificados de Amazon

IAM función	Compatibilidad con Verified Permissions
Políticas basadas en identidades	Sí
Políticas basadas en recursos	No
Acciones de políticas	Sí
Recursos de políticas	Sí
Claves de condición de política	No
ACLs	No
ABAC (etiquetas en políticas)	Sí
Credenciales temporales	Sí
Permisos de entidades principales	Sí
Roles de servicio	No
Roles vinculados al servicio	No

Para obtener una visión general de cómo funcionan los permisos verificados y otros AWS servicios con la mayoría de IAM las funciones, consulte AWS los servicios con los que funcionan IAM en la Guía del IAM usuario.

Políticas basadas en identidad para Verified Permissions

Compatibilidad con las políticas basadas en identidad	Sí

Las políticas basadas en identidad son documentos de políticas de permisos JSON que puede asociar a una identidad, como un usuario de IAM, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener información sobre cómo crear una política basada en la identidad, consulte Definir IAM permisos personalizados con políticas administradas por el cliente en la Guía del IAM usuario.

Con las políticas IAM basadas en la identidad, puede especificar las acciones y los recursos permitidos o denegados, así como las condiciones en las que se permiten o deniegan las acciones. Para obtener más información acerca de los elementos que puede utilizar en una política de JSON, consulte Referencia de los elementos de las políticas de JSON de IAM en la Guía del usuario de IAM .

Ejemplos de políticas basadas en identidad para Verified Permissions

Para ver ejemplos de políticas basadas en identidad de Verified Permissions, consulte Ejemplos de políticas basadas en identidad para Amazon Verified Permissions.

Políticas basadas en recursos de Verified Permissions

Compatibilidad con las políticas basadas en recursos	No

Las políticas basadas en recursos son documentos de política JSON que se asocian a un recurso. Algunos ejemplos de políticas basadas en recursos son las políticas de confianza en los IAM roles y las políticas de compartimentos. Amazon S3 En los servicios que admiten políticas basadas en recursos, los administradores de servicios pueden utilizarlos para controlar el acceso a un recurso específico. Para el recurso al que se asocia la política, la política define qué acciones puede realizar una entidad principal especificada en ese recurso y en qué condiciones. Debe especificar una entidad principal en una política basada en recursos. Los principales pueden incluir cuentas, usuarios, roles, usuarios federados o. Servicios de AWS

Para habilitar el acceso entre cuentas, puede especificar una cuenta completa o IAM entidades de otra cuenta como principales en una política basada en recursos. Para obtener más información, consulte el acceso a los recursos entre cuentas IAM en la Guía del IAM usuario.

Acciones de política para Verified Permissions

Admite acciones de política	Sí

Los administradores pueden usar las políticas de AWS JSON para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Action` de una política JSON describe las acciones que puede utilizar para conceder o denegar el acceso en una política. Incluya acciones en una política para conceder permisos y así llevar a cabo la operación asociada.

Para ver una lista de las acciones de Verified Permissions, consulte Acciones definidas por Amazon Verified Permissions en la Referencia de autorizaciones de servicio.

Las acciones de políticas de Verified Permissions utilizan el siguiente prefijo antes de la acción:

```
verifiedpermissions
```

Para especificar varias acciones en una única instrucción, sepárelas con comas.

```
"Action": [
      "verifiedpermissions:action1",
      "verifiedpermissions:action2"
         ]
```

Puede utilizar caracteres comodín (*) para especificar varias acciones . Por ejemplo, para especificar todas las acciones que comiencen con la palabra `Get`, incluya la siguiente acción:

```
"Action": "verifiedpermissions:Get*"
```

Para ver ejemplos de políticas basadas en identidad de Verified Permissions, consulte Ejemplos de políticas basadas en identidad para Amazon Verified Permissions.

Recursos de políticas para Verified Permissions

Admite recursos de políticas	Sí

Los administradores pueden usar las políticas de AWS JSON para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Resource` de la política JSON especifica el objeto u objetos a los que se aplica la acción. Como práctica recomendada, especifique un recurso utilizando el Nombre de recurso de Amazon (ARN). En el caso de las acciones que no admiten permisos por recurso, utilice un carácter comodín (*) para indicar que la instrucción se aplica a todos los recursos.

```
"Resource": "*"
```

Para ver una lista de los tipos de recursos de permisos verificados y sus correspondientes ARNs, consulte Tipos de recursos definidos por los permisos verificados de Amazon en la Referencia de autorización de servicio. Para obtener información sobre las acciones con las que puede especificar el ARN de cada recurso, consulte Acciones definidas por Amazon Verified Permissions.

Claves de condición de política de Amazon Verified Permissions

Admite claves de condición de políticas específicas del servicio	No

Los administradores pueden usar las políticas de AWS JSON para especificar quién tiene acceso a qué. Es decir, qué entidad principal puede realizar acciones en qué recursos y en qué condiciones.

El elemento `Condition` especifica cuándo se ejecutan las instrucciones en función de criterios definidos. Puede crear expresiones condicionales que utilizan operadores de condición, tales como igual o menor que, para que la condición de la política coincida con los valores de la solicitud. Para ver todas las claves de condición AWS globales, consulte las claves de contexto de condición AWS globales en la Guía del IAM usuario.

ACLs en Permisos verificados

Soporta ACLs	No

Las listas de control de acceso (ACLs) controlan qué directores (miembros de la cuenta, usuarios o roles) tienen permisos para acceder a un recurso. ACLs son similares a las políticas basadas en recursos, aunque no utilizan el formato de documento de políticas JSON.

ABAC con Verified Permissions

Admite ABAC (etiquetas en las políticas)	Sí

El control de acceso basado en atributos (ABAC) es una estrategia de autorización que define permisos en función de atributos denominados etiquetas. Puede adjuntar etiquetas a IAM las entidades y AWS los recursos y, a continuación, diseñar políticas de ABAC para permitir las operaciones cuando la etiqueta del principal coincida con la etiqueta del recurso.

Para controlar el acceso en función de etiquetas, debe proporcionar información de las etiquetas en el elemento de condición de una política utilizando las claves de condición `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

Si un servicio admite las tres claves de condición para cada tipo de recurso, el valor es Sí para el servicio. Si un servicio admite las tres claves de condición solo para algunos tipos de recursos, el valor es Parcial.

Para obtener más información sobre el ABAC, consulte Definir permisos con la autorización del ABAC en la Guía del IAM usuario. Para ver un tutorial con los pasos para configurar ABAC, consulte Uso del control de acceso basado en atributos (ABAC) en la Guía del usuario de IAM .

Uso de credenciales temporales con Verified Permissions

Compatible con el uso de credenciales temporales	Sí

Las credenciales temporales proporcionan acceso a AWS los recursos a corto plazo y se crean automáticamente cuando se utiliza la federación o se cambia de rol. AWS recomienda generar credenciales temporales de forma dinámica en lugar de utilizar claves de acceso a largo plazo. Para obtener más información, consulte las credenciales de seguridad temporales de IAM y con las Servicios de AWS que funcionan IAM en la Guía del IAM usuario.

Permisos de entidades principales entre servicios de Verified Permissions

Admite permisos de entidades principales	Sí

Las sesiones de acceso directo (FAS) utilizan los permisos de la persona principal que realiza la llamada Servicio de AWS, además de los que solicitan, Servicio de AWS para realizar solicitudes a los servicios descendentes. Para obtener información sobre las políticas a la hora de realizar solicitudes de FAS, consulte Sesiones de acceso directo.

Roles de servicio para Verified Permissions

Compatible con roles de servicio	No

Un rol de servicio es un rol de IAM que asume un servicio para realizar acciones en su nombre. Un IAM administrador puede crear, modificar y eliminar un rol de servicio desde dentro IAM. Para obtener más información, consulte Crear un rol para delegar permisos a un Servicio de AWS en la Guía del IAM usuario.

Permisos de roles vinculados al servicio para Verified Permissions

Compatible con roles vinculados al servicio	No

Un rol vinculado a un servicio es un tipo de rol de servicio que está vinculado a un. Servicio de AWS El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados al servicio aparecen en usted Cuenta de AWS y son propiedad del servicio. Un IAM administrador puede ver los permisos de los roles vinculados al servicio, pero no editarlos.

Para obtener más información sobre la creación o la administración de funciones vinculadas a un servicio, consulte los AWS servicios que funcionan con. IAM Busque un servicio en la tabla que incluya `Yes` en la columna Rol vinculado a un servicio. Seleccione el vínculo Sí para ver la documentación acerca del rol vinculado a servicios para ese servicio.

IAM políticas de permisos verificados

Verified Permissions administra los permisos de los usuarios dentro de la aplicación. Para que su aplicación pueda acceder a los permisos verificados APIs o para que Consola de administración de AWS los usuarios puedan gestionar las políticas de Cedar en un almacén de políticas de permisos verificados, debe añadir los IAM permisos necesarios.

Las políticas basadas en la identidad son documentos de política de permisos de JSON que puede adjuntar a una identidad, como un IAM usuario, un grupo de usuarios o un rol. Estas

políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener información sobre cómo crear una política basada en la identidad, consulte Creación de IAM políticas en la Guía del usuario. IAM

Con las políticas IAM basadas en la identidad, puede especificar las acciones y los recursos permitidos o denegados, así como las condiciones en las que se permiten o deniegan las acciones (enumeradas a continuación). No es posible especificar la entidad principal en una política basada en identidad porque se aplica al usuario o rol al que está asociada. Para obtener más información sobre todos los elementos que puede utilizar en una política de JSON, consulte la referencia sobre los elementos de la política de IAM JSON en la Guía del IAM usuario.

Action	Descripción
CreateIdentitySource	Acción para crear una nueva fuente de identidad.
CreatePolicy	Acción para crear una política de Cedar en un almacén de políticas. Puede crear una política estática o una política vinculada a una plantilla de política.
CreatePolicyStore	Acción para crear un nuevo almacén de políticas.
CreatePolicyTemplate	Acción para crear una nueva plantilla de política.
DeleteIdentitySource	Acción para eliminar una fuente de identidad.
DeletePolicy	Acción para eliminar una política de un almacén de políticas.
DeletePolicyStore	Acción para eliminar un almacén de políticas.
DeletePolicyTemplate	Acción para eliminar una plantilla de política.
GetIdentitySource	Acción para obtener una fuente de identidad.

Action	Descripción
GetPolicy	Acción para recuperar información sobre una política específica.
GetPolicyStore	Acción para recuperar información sobre un almacén de políticas específico.
GetPolicyTemplate	Acción para obtener una plantilla de política.
GetSchema	Acción para obtener un esquema.
IsAuthorized	Acción para obtener una respuesta de autorización en función de los parámetros descritos en la solicitud de autorización.
IsAuthorizedWithToken	Acción para obtener una respuesta de autorización basada en los parámetros descritos en la solicitud de autorización, donde el principal proviene de un token de identidad.
ListIdentitySources	Acción para enumerar todas las fuentes de identidad del Cuenta de AWS.
ListPolicies	Acción para enumerar todas las políticas de un almacén de políticas.
ListPolicyStores	Acción para enumerar todos los almacenes de políticas del Cuenta de AWS.
ListPolicyTemplates	Acción para enumerar todas las plantillas de políticas en Cuenta de AWS.
ListTagsForResource	Acción para enumerar todas las etiquetas de un recurso.
PutSchema	Acción para añadir un esquema a un almacén de políticas.
TagResource	Acción para añadir una etiqueta a un recurso.

Action	Descripción
UpdateIdentitySource	Acción para actualizar una fuente de identidad.
UpdatePolicy	Acción para actualizar una política en un almacén de políticas.
UpdatePolicyStore	Acción para actualizar un almacén de políticas.
UpdatePolicyTemplate	Acción para actualizar una plantilla de políticas.
UntagResource	Acción para eliminar una etiqueta de un recurso.

Ejemplo IAM de política de autorización para la CreatePolicy acción:

JSON

```
{
"Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "verifiedpermissions:CreatePolicy"
            ],
            "Resource": "*"
        }
    ]
 }
```

Ejemplos de políticas basadas en identidad para Amazon Verified Permissions

De forma predeterminada, los usuarios y roles no tienen permiso para crear, ver ni modificar recursos de Verified Permissions. Tampoco pueden realizar tareas mediante la Consola de administración de AWS, AWS Command Line Interface (AWS CLI) o la AWS API. IAM El administrador debe crear IAM

políticas que concedan permiso a los usuarios y a los roles para realizar acciones en los recursos que necesiten. El administrador debe asociar esas políticas a los usuarios que las necesiten.

Para obtener información sobre cómo crear una política IAM basada en la identidad mediante estos ejemplos de documentos de política de JSON, consulte Creación de IAM políticas en la Guía del IAM usuario.

Para obtener más información sobre las acciones y los tipos de recursos definidos por los permisos verificados, incluido el ARNs formato de cada uno de los tipos de recursos, consulte Acciones, recursos y claves de condición de los permisos verificados de Amazon en la Referencia de autorización de servicios.

Temas

- Prácticas recomendadas sobre las políticas
- Uso de la consola de Verified Permissions
- Cómo permitir a los usuarios consultar sus propios permisos

Prácticas recomendadas sobre las políticas

Las políticas basadas en identidades determinan si alguien puede crear, acceder o eliminar los recursos de Verified Permissions en su cuenta. Estas acciones pueden generar costos adicionales para su Cuenta de AWS. Siga estas directrices y recomendaciones al crear o editar políticas basadas en identidades:

- Comience con las políticas AWS administradas y avance hacia los permisos con privilegios mínimos: para empezar a conceder permisos a sus usuarios y cargas de trabajo, utilice las políticas AWS administradas que otorgan permisos en muchos casos de uso comunes. Están disponibles en su. Cuenta de AWS Le recomendamos que reduzca aún más los permisos definiendo políticas administradas por el AWS cliente que sean específicas para sus casos de uso. Con el fin de obtener más información, consulte las políticas administradas por AWS o las políticas administradas por AWS para funciones de trabajo en la Guía de usuario de IAM .
- Aplique permisos con privilegios mínimos: cuando establezca permisos con IAM políticas, conceda solo los permisos necesarios para realizar una tarea. Para ello, debe definir las acciones que se pueden llevar a cabo en determinados recursos en condiciones específicas, también conocidos como permisos de privilegios mínimos. Para obtener más información sobre cómo IAM aplicar permisos, consulte Políticas y permisos IAM en la IAM Guía del usuario.

- Utilice las condiciones en IAM las políticas para restringir aún más el acceso: puede añadir una condición a sus políticas para limitar el acceso a las acciones y los recursos. Por ejemplo, puede escribir una condición de políticas para especificar que todas las solicitudes deben enviarse utilizando SSL. También puedes usar condiciones para conceder el acceso a las acciones del servicio si se utilizan a través de una acción específica Servicio de AWS, por ejemplo CloudFormation. Para obtener más información, consulte Elementos de la política JSON de IAM : condición en la Guía del usuario de IAM .
- Utilice IAM Access Analyzer para validar sus IAM políticas y garantizar permisos seguros y funcionales. IAM Access Analyzer valida las políticas nuevas y existentes para que se ajusten al lenguaje de políticas (JSON) y a las IAM mejores prácticas de IAM. El analizador de acceso de IAM proporciona más de 100 verificaciones de políticas y recomendaciones procesables para ayudar a crear políticas seguras y funcionales. Para obtener más información, consulte Validar políticas con IAM Access Analyzer en la Guía del usuario.IAM
- Requerir autenticación multifactor (MFA): si tiene un escenario que requiere usuarios de IAM o un usuario raíz en Cuenta de AWS su cuenta, active la MFA para mayor seguridad. Para exigir la MFA cuando se invoquen las operaciones de la API, añada condiciones de MFA a sus políticas. Para obtener más información, consulte Acceso seguro a la API con MFA en la Guía del IAM usuario.

Para obtener más información sobre las prácticas recomendadas IAM, consulte las prácticas recomendadas de seguridad IAM en la Guía del IAM usuario.

Uso de la consola de Verified Permissions

Para acceder a la consola de Amazon Verified Permissions, debe tener un conjunto mínimo de permisos. Estos permisos deben permitirle enumerar y ver detalles sobre los recursos de permisos verificados de su cuenta Cuenta de AWS. Si crea una política basada en identidades que sea más restrictiva que el mínimo de permisos necesarios, la consola no funcionará del modo esperado para las entidades (usuarios o roles) que tengan esa política.

No es necesario que concedas permisos mínimos de consola a los usuarios que solo realicen llamadas a la API AWS CLI o a la AWS API. En su lugar, permita el acceso únicamente a las acciones que coincidan con la operación de API que intentan realizar.

Para garantizar que los usuarios y los roles puedan seguir utilizando la consola de permisos verificados, adjunta también los permisos verificados `ConsoleAccess` o la política `ReadOnly`

AWS gestionada a las entidades. Para obtener más información, consulte Adición de permisos a un usuario en la Guía del usuario de IAM .

Cómo permitir a los usuarios consultar sus propios permisos

En este ejemplo, se muestra cómo podría crear una política que permita a los usuarios de IAM ver las políticas administradas e insertadas que se asocian a la identidad de sus usuarios. Esta política incluye permisos para completar esta acción en la consola o mediante programación mediante la API AWS CLI o AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS políticas gestionadas para los permisos verificados de Amazon

Para añadir permisos a usuarios, grupos y roles, es más fácil usar políticas AWS administradas que escribirlas tú mismo. Crear políticas gestionadas por los IAM clientes que proporcionen a tu equipo solo los permisos que necesita requiere tiempo y experiencia. Para empezar rápidamente, puedes usar nuestras políticas AWS gestionadas. Estas políticas cubren casos de uso comunes y están disponibles en su Cuenta de AWS. Para obtener más información sobre las políticas AWS administradas, consulte las políticas AWS administradas en la Guía del IAM usuario.

AWS los servicios mantienen y AWS actualizan las políticas administradas. No puede cambiar los permisos en las políticas AWS gestionadas. En ocasiones, los servicios agregan permisos adicionales a una política administrada de AWS para admitir características nuevas. Este tipo de actualización afecta a todas las identidades (usuarios, grupos y roles) donde se asocia la política. Es más probable que los servicios actualicen una política administrada de AWS cuando se lanza una nueva característica o cuando se ponen a disposición nuevas operaciones. Los servicios no eliminan los permisos de una política AWS administrada, por lo que las actualizaciones de la política no afectarán a los permisos existentes.

Además, AWS admite políticas administradas para funciones laborales que abarcan varios servicios. Por ejemplo, la política ReadOnlyAccess AWS gestionada proporciona acceso de solo lectura a todos los AWS servicios y recursos. Cuando un servicio lanza una nueva función, AWS agrega permisos de solo lectura para nuevas operaciones y recursos. Para obtener una lista y una descripción de las políticas de funciones laborales, consulte las políticas AWS gestionadas para las funciones laborales en la Guía del IAM usuario.

AWS política gestionada: AmazonVerifiedPermissionsFullAccess

La política `AmazonVerifiedPermissionsFullAccess` gestionada otorga acceso total a los permisos verificados. Para trabajar con fuentes de identidad Amazon Cognito basadas en datos, tendrás que adjuntar una política independiente, como la AmazonCognitoReadOnlypolítica.

JSON

```
{
```

```
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "AccountLevelPermissions",
      "Effect": "Allow",
      "Action": [
        "verifiedpermissions:CreatePolicyStore",
        "verifiedpermissions:ListPolicyStores"
      ],
      "Resource": "*"
    },
    {
      "Sid": "PolicyStoreLevelPermissions",
      "Effect": "Allow",
      "Action": [
        "verifiedpermissions:*"
      ],
      "Resource": [
        "arn:aws:verifiedpermissions::*:policy-store/*"
      ]
    }
  ]
}
```

AWS política gestionada: AmazonVerifiedPermissionsReadOnlyAccess

La política `AmazonVerifiedPermissionsReadOnlyAccess` gestionada concede acceso de solo lectura a los permisos verificados.

Esta política otorga acceso a todas las operaciones de lectura de Amazon Verified Permissions, incluida la consulta de autorización APIs `IsAuthorized` y`IsAuthorizedWithToken`.

> **Note**
>
> El acceso `BatchIsAuthorized` y `BatchIsAuthorizedWithToken` se conceden automáticamente cuando se concede el acceso a `IsAuthorized` y`IsAuthorizedWithToken`, respectivamente.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "AccountLevelPermissions",
      "Effect": "Allow",
      "Action": [
        "verifiedpermissions:ListPolicyStores"
      ],
      "Resource": "*"
    },
    {
      "Sid": "PolicyStoreLevelPermissions",
      "Effect": "Allow",
      "Action": [
        "verifiedpermissions:GetIdentitySource",
        "verifiedpermissions:GetPolicy",
        "verifiedpermissions:GetPolicyStore",
        "verifiedpermissions:GetPolicyTemplate",
        "verifiedpermissions:GetSchema",
        "verifiedpermissions:IsAuthorized",
        "verifiedpermissions:IsAuthorizedWithToken",
        "verifiedpermissions:ListIdentitySources",
        "verifiedpermissions:ListPolicies",
        "verifiedpermissions:ListPolicyTemplates"
      ],
      "Resource": [
        "arn:aws:verifiedpermissions::*:policy-store/*"
      ]
    }
  ]
}
```

Permisos verificados: actualizaciones de las políticas AWS administradas

Consulta los detalles sobre las actualizaciones de las políticas AWS administradas para los permisos verificados desde que este servicio comenzó a rastrear estos cambios. Para recibir alertas

automáticas sobre los cambios en esta página, suscríbase a la fuente RSS de la página del historial del documento de permisos verificados.

Cambio	Descripción	Fecha
AmazonVerifiedPermissionsFullAccess: política nueva	Verified Permissions agregó una nueva política para permitir el acceso total a los permisos verificados.	11 de octubre de 2024
AmazonVerifiedPermissionsReadOnlyAccess: política nueva	Verified Permissions agregó una nueva política para permitir el acceso a todas las operaciones de lectura de Amazon Verified Permissions, incluida la consulta de autorización APIs `IsAuthorized` y `IsAuthorizedWithToken` .	11 de octubre de 2024
Verified Permissions comenzó a rastrear los cambios	Verified Permissions comenzó a realizar un seguimiento de los cambios en sus políticas AWS gestionadas.	11 de octubre de 2024

Solución de problemas de identidad y acceso de Amazon Verified Permissions

Utilice la siguiente información para diagnosticar y solucionar los problemas comunes que puedan surgir cuando trabaje con Verified Permissions e IAM.

Temas

- No tengo autorización para realizar una acción en Verified Permissions
- No estoy autorizado a realizar tareas como: PassRole

- Quiero permitir que personas ajenas a mí accedan Cuenta de AWS a mis recursos de permisos verificados

No tengo autorización para realizar una acción en Verified Permissions

Si recibe un error que indica que no tiene autorización para realizar una acción, las políticas se deben actualizar para permitirle realizar la acción.

En el siguiente ejemplo, el error se produce cuando el usuario de IAM `mateojackson` intenta utilizar la consola para consultar los detalles acerca de un recurso ficticio *`my-example-widget`*, pero no tiene los permisos ficticios `verifiedpermissions:`*`GetWidget`*.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 verifiedpermissions:GetWidget on resource: my-example-widget
```

En este caso, la política del usuario `mateojackson` debe actualizarse para permitir el acceso al recurso *`my-example-widget`* mediante la acción `verifiedpermissions:`*`GetWidget`*.

Si necesita ayuda, póngase en contacto con su AWS administrador. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

No estoy autorizado a realizar tareas como: PassRole

Si recibe un error que indica que no tiene autorización para realizar la acción `iam:PassRole`, se deben actualizar las políticas a fin de permitirle pasar un rol a Verified Permissions.

Algunas Servicios de AWS permiten transferir una función existente a ese servicio en lugar de crear una nueva función de servicio o una función vinculada a un servicio. Para ello, debe tener permisos para transferir la función al servicio.

En el siguiente ejemplo, el error se produce cuando un usuario de IAM denominado `marymajor` intenta utilizar la consola para realizar una acción en Verified Permissions. Sin embargo, la acción requiere que el servicio cuente con permisos que otorguen un rol de servicio. Mary no tiene permisos para transferir el rol al servicio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

En este caso, las políticas de Mary se deben actualizar para permitirle realizar la acción `iam:PassRole`.

Si necesita ayuda, póngase en contacto con su AWS administrador. El administrador es la persona que le proporcionó las credenciales de inicio de sesión.

Quiero permitir que personas ajenas a mí accedan Cuenta de AWS a mis recursos de permisos verificados

Se puede crear un rol que los usuarios de otras cuentas o las personas externas a la organización puedan utilizar para acceder a sus recursos. Se puede especificar una persona de confianza para que asuma el rol. En el caso de los servicios que admiten políticas basadas en recursos o listas de control de acceso (ACLs), puedes usar esas políticas para permitir que las personas accedan a tus recursos.

Para obtener más información, consulte lo siguiente:

- Para obtener información acerca de si Verified Permissions admite estas características, consulte Cómo funciona Amazon Verified Permissions con IAM.
- Para obtener información sobre cómo proporcionar acceso a los recursos de su Cuentas de AWS propiedad, consulte Proporcionar acceso a un usuario de IAM en otro usuario de su propiedad Cuenta de AWS en la Guía del IAM usuario.
- Para obtener información sobre cómo proporcionar acceso a sus recursos a terceros Cuentas de AWS, consulte Proporcionar acceso a recursos que Cuentas de AWS son propiedad de terceros en la Guía del IAM usuario.
- Para obtener información sobre cómo proporcionar acceso mediante una federación de identidades, consulte Proporcionar acceso a usuarios autenticadoPara que su aplicación pueda acceder a s externamente (federación de identidades) en la Guía del usuario de IAM .
- Para saber la diferencia entre usar roles y políticas basadas en recursos para el acceso entre cuentas, consulta el tema Acceso a recursos entre cuentas IAM en la Guía del IAM usuario.

Validación de conformidad de Amazon Verified Permissions

Nuestra nueva experiencia AWS de registro no está diseñada para cargas de trabajo reguladas. Si utilizas nuestra nueva experiencia de AWS registro, pero quieres utilizarla AWS para cargas de trabajo reguladas, puedes suscribirte a una versión AWS (avanzada) o activar funciones avanzadas para tu entorno. AWS

Para saber si una Servicio de AWS está dentro del ámbito de un programa de cumplimiento específico, consulte Servicios de AWS la sección Alcance por programa de cumplimiento y

seleccione el programa de cumplimiento que le interese. Para obtener información general, consulte Programas AWS de cumplimiento Programas .

Puede descargar informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte Descargar informes en AWS Artifact .

Su responsabilidad de cumplimiento al usarlos Servicios de AWS viene determinada por la confidencialidad de sus datos, los objetivos de cumplimiento de su empresa y las leyes y normativas aplicables. Para obtener más información sobre su responsabilidad de cumplimiento al usarlos Servicios de AWS, consulte la documentación AWS de seguridad.

Resiliencia de Amazon Verified Permissions

La infraestructura AWS global se basa en Regiones de AWS zonas de disponibilidad. Regiones de AWS proporcionan varias zonas de disponibilidad aisladas y separadas físicamente, que están conectadas mediante redes de baja latencia, alto rendimiento y alta redundancia. Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre las zonas sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de uno o varios centros de datos.

Al crear un almacén de políticas de permisos verificados, se crea dentro de una persona Región de AWS y se replica automáticamente en los centros de datos que componen las zonas de disponibilidad de esa región. En este momento, Verified Permissions no admite ninguna replicación entre regiones.

Para obtener más información sobre las zonas de disponibilidad Regiones de AWS y las zonas de disponibilidad, consulte Infraestructura AWS global.

Supervisión de las llamadas a la API de permisos verificados de Amazon

La supervisión es una parte importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de Amazon Verified Permissions y del resto de sus AWS soluciones. AWS proporciona las siguientes herramientas para supervisar los permisos verificados, informar cuando algo va mal y tomar medidas automáticas cuando sea necesario:

- AWS CloudTrailcaptura las llamadas a la API y los eventos relacionados realizados por tu AWS cuenta o en su nombre y envía los archivos de registro a un Amazon S3 depósito que tú especifiques. Puedes identificar qué usuarios y cuentas llamaron AWS, la dirección IP de origen desde la que se realizaron las llamadas y cuándo se produjeron. Para obtener más información, consulte la Guía del usuario de AWS CloudTrail.

Para obtener más información sobre cómo supervisar los permisos verificados con CloudTrail, consulteRegistrar las llamadas a la API de permisos verificados de Amazon mediante AWS CloudTrail.

Registrar las llamadas a la API de permisos verificados de Amazon mediante AWS CloudTrail

Los permisos verificados de Amazon están integrados con AWS CloudTrail un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o un AWS servicio en los permisos verificados. CloudTrail captura todas las llamadas a la API para obtener permisos verificados como eventos. Las llamadas capturadas incluyen las llamadas realizadas desde la consola de Verified Permissions y las llamadas de código a las operaciones de la API de Verified Permissions. Si creas un registro, puedes habilitar la entrega continua de CloudTrail eventos a un Amazon S3 bucket, incluidos los eventos para los permisos verificados. Si no configuras un registro, puedes seguir viendo los eventos de acción de administración más recientes en la CloudTrail consola, en el historial de eventos, pero no los eventos de las llamadas a la API, como por ejemplo**`isAuthorized`**. Con la información recopilada por CloudTrail, puedes determinar la solicitud que se hizo a Verified Permissions, la dirección IP desde la que se hizo la solicitud, quién la hizo, cuándo se hizo y otros detalles.

Para obtener más información CloudTrail, consulta la Guía AWS CloudTrail del usuario.

La información de permisos verificados se encuentra en CloudTrail

CloudTrail se habilita Cuenta de AWS cuando creas la cuenta. Cuando se produce actividad en los permisos verificados, esa actividad se registra en un CloudTrail evento junto con otros eventos de AWS servicio en el historial de eventos. Puede ver, buscar y descargar eventos recientes en su Cuenta de AWS. Para obtener más información, consulta Visualización de eventos con el historial de CloudTrail eventos.

Para obtener un registro continuo de tus eventos Cuenta de AWS, incluidos los eventos de los permisos verificados, crea un registro. Un registro permite CloudTrail enviar archivos de registro a un Amazon S3 bucket. De forma predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las Regiones de AWS. El registro registra los eventos de todas las regiones de la AWS partición y envía los archivos de registro al Amazon S3 bucket que usted especifique. Además, puede configurar otros AWS servicios para analizar más a fondo los datos de eventos recopilados en los CloudTrail registros y actuar en función de ellos. Para más información, consulte los siguientes temas:

- Introducción a la creación de registros de seguimiento
- CloudTrail servicios e integraciones compatibles
- Configurar las notificaciones de Amazon SNS para CloudTrail
- Recibir archivos de CloudTrail registro de varias regiones y Recibir archivos de CloudTrail registro de varias cuentas

Todas las acciones relacionadas con los permisos verificados se registran CloudTrail y documentan en la Guía de referencia de la API de permisos verificados de Amazon. Por ejemplo, las llamadas a `CreateIdentitySourceDeletePolicy`, y `ListPolicyStores` las acciones generan entradas en los archivos de CloudTrail registro.

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario le ayuda a determinar lo siguiente:

- Si la solicitud se realizó con credenciales de usuario root o AWS Identity and Access Management (IAM).
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud fue realizada por otro AWS servicio.

Para obtener más información, consulte el elemento userIdentity de CloudTrail .

Los eventos de datos, por ejemplo IsAuthorized, no IsAuthorizedWithToken se registran de forma predeterminada al crear un banco de datos de rutas o eventos. Para registrar CloudTrail los eventos de datos, debes agregar de forma explícita los recursos compatibles o los tipos de recursos para los que deseas recopilar la actividad. Para obtener más información, consulte Eventos de datos en la Guía del usuario de AWS CloudTrail .

Descripción de las entradas del archivo de registro de Verified Permissions

Un registro es una configuración que permite la entrega de eventos como archivos de registro a un Amazon S3 bucket que especifiques. CloudTrail Los archivos de registro contienen una o más entradas de registro. Un evento representa una única solicitud de cualquier fuente e incluye información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. CloudTrail los archivos de registro no son un seguimiento ordenado de las llamadas a la API pública, por lo que no aparecen en ningún orden específico.

En el caso de las llamadas a la API de autorización, los elementos de respuesta, como la decisión, se incluyen `additionalEventData` en lugar de`responseElements`.

Temas

- IsAuthorized
- IsAuthorizedWithToken
- BatchIsAuthorized
- BatchIsAuthorizedWithToken
- CreatePolicyStore
- ListPolicyStores
- DeletePolicyStore
- PutSchema
- GetSchema
- CreatePolicyTemplate
- DeletePolicyTemplate
- CreatePolicy
- GetPolicy
- CreateIdentitySource

- GetIdentitySource
- ListIdentitySources
- DeleteIdentitySource

> **Note**
>
> Algunos campos se han eliminado de los ejemplos por motivos de privacidad de datos.

IsAuthorized

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `IsAuthorized` API:


```
{
    "eventVersion": "1.08",
    "userIdentity": {
  "type": "AssumedRole",
  "principalId": "EXAMPLE_PRINCIPAL_ID",
  "arn": "arn:aws:iam::123456789012:role/ExampleRole",
  "accountId": "123456789012",
  "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
    },
    "eventTime": "2023-11-20T22:55:03Z",
    "eventSource": "verifiedpermissions.amazonaws.com",
    "eventName": "IsAuthorized",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "203.0.113.0",
    "userAgent": "aws-cli/2.11.18 Python/3.11.3 Linux/5.4.241-160.348.amzn2int.x86_64
 exe/x86_64.amzn.2 prompt/off command/verifiedpermissions.is-authorized",
    "requestParameters": {
        "principal": {
            "entityType": "PhotoFlash::User",
            "entityId": "alice"
        },
        "action": {
            "actionType": "PhotoFlash::Action",
            "actionId": "ViewPhoto"
        },
        "resource": {
            "entityType": "PhotoFlash::Photo",
```

```
            "entityId": "VacationPhoto94.jpg"
        },
        "policyStoreId": "PSEXAMPLEabcdefg111111"
    },
    "responseElements": null,
    "additionalEventData": {
        "decision": "ALLOW"
    },
    "requestID": "346c4b6a-d12f-46b6-bc06-6c857bd3b28e",
    "eventID": "8a4fed32-9605-45dd-a09a-5ebbf0715bbc",
    "readOnly": true,
    "resources": [
        {
    "accountId": "123456789012",
    "type": "AWS::VerifiedPermissions::PolicyStore",
    "ARN": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data"
}
```

IsAuthorizedWithToken

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `IsAuthorizedWithToken` API:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
  "type": "AssumedRole",
  "principalId": "EXAMPLE_PRINCIPAL_ID",
  "arn": "arn:aws:iam::123456789012:role/ExampleRole",
  "accountId": "123456789012",
  "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
    },
    "eventTime": "2023-11-20T22:55:03Z",
    "eventSource": "verifiedpermissions.amazonaws.com",
    "eventName": "IsAuthorizedWithToken",
    "awsRegion": "us-west-2",
```

```
    "sourceIPAddress": "203.0.113.0",
    "userAgent": "aws-cli/2.11.18 Python/3.11.3 Linux/5.4.241-160.348.amzn2int.x86_64
 exe/x86_64.amzn.2 prompt/off command/verifiedpermissions.is-authorized-with-token",
    "requestParameters": {
        "policyStoreId": "PSEXAMPLEabcdefg111111",
        "resource": {
            "entityType": "PhotoFlash::Photo",
            "entityId": "VacationPhoto94.jpg"
        },
        "action": {
            "actionType": "PhotoFlash::Action",
            "actionId": "ViewPhoto"
        }
    },
    "responseElements": null,
    "additionalEventData": {
        "decision": "ALLOW",
        "principal": {
            "entityType": "PhotoFlash::User",
            "entityId": "us-east-1_EXAMPLE|EXAMPLE_SUBJECT_ID"
        }
    },
    "requestID": "346c4b6a-d12f-46b6-bc06-6c857bd3b28e",
    "eventID": "8a4fed32-9605-45dd-a09a-5ebbf0715bbc",
    "readOnly": true,
    "resources": [
        {
    "accountId": "123456789012",
    "type": "AWS::VerifiedPermissions::PolicyStore",
    "ARN": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data"
}
```


BatchIsAuthorized

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `BatchIsAuthorized` API:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
  "type": "AssumedRole",
  "principalId": "EXAMPLE_PRINCIPAL_ID",
  "arn": "arn:aws:iam::123456789012:role/ExampleRole",
  "accountId": "123456789012",
  "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
    },
    "eventTime": "2023-11-20T23:02:33Z",
    "eventSource": "verifiedpermissions.amazonaws.com",
    "eventName": "BatchIsAuthorized",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "203.0.113.0",
    "userAgent": "aws-cli/2.11.18 Python/3.11.3 Linux/5.4.241-160.348.amzn2int.x86_64
 exe/x86_64.amzn.2 prompt/off command/verifiedpermissions.is-authorized",
    "requestParameters": {
        "requests": [
            {
                "principal": {
                    "entityType": "PhotoFlash::User",
                    "entityId": "alice"
                },
                "action": {
                    "actionType": "PhotoFlash::Action",
                    "actionId": "ViewPhoto"
                },
                "resource": {
                    "entityType": "PhotoFlash::Photo",
                    "entityId": "VacationPhoto94.jpg"
                }
            },
            {
                "principal": {
                    "entityType": "PhotoFlash::User",
                    "entityId": "annalisa"
                },
                "action": {
                    "actionType": "PhotoFlash::Action",
                    "actionId": "DeletePhoto"
                },
                "resource": {
                    "entityType": "PhotoFlash::Photo",
```

```
                    "entityId": "VacationPhoto94.jpg"
                }
            }
        ],
        "policyStoreId": "PSEXAMPLEabcdefg111111"
    },
    "responseElements": null,
    "additionalEventData": {
        "results": [
            {
                "request": {
                    "principal": {
                        "entityType": "PhotoFlash::User",
                        "entityId": "alice"
                    },
                    "action": {
                        "actionType": "PhotoFlash::Action",
                        "actionId": "ViewPhoto"
                    },
                    "resource": {
                        "entityType": "PhotoFlash::Photo",
                        "entityId": "VacationPhoto94.jpg"
                    }
                },
                "decision": "ALLOW"
            },
            {
                "request": {
                    "principal": {
                        "entityType": "PhotoFlash::User",
                        "entityId": "annalisa"
                    },
                    "action": {
                        "actionType": "PhotoFlash::Action",
                        "actionId": "DeletePhoto"
                    },
                    "resource": {
                        "entityType": "PhotoFlash::Photo",
                        "entityId": "VacationPhoto94.jpg"
                    }
                },
                "decision": "DENY"
            }
        ]
```

```
    },
    "requestID": "a8a5caf3-78bd-4139-924c-7101a8339c3b",
    "eventID": "7d81232f-f3d1-4102-b9c9-15157c70487b",
    "readOnly": true,
    "resources": [
        {
    "accountId": "123456789012",
    "type": "AWS::VerifiedPermissions::PolicyStore",
    "ARN": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data"
}
```

BatchIsAuthorizedWithToken

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `BatchIsAuthorizedWithToken` API:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
  "type": "AssumedRole",
  "principalId": "EXAMPLE_PRINCIPAL_ID",
  "arn": "arn:aws:iam::123456789012:role/ExampleRole",
  "accountId": "123456789012",
  "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
    },
    "eventTime": "2023-11-20T23:02:33Z",
    "eventSource": "verifiedpermissions.amazonaws.com",
    "eventName": "BatchIsAuthorizedWithToken",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "203.0.113.0",
    "userAgent": "aws-cli/2.11.18 Python/3.11.3 Linux/5.4.241-160.348.amzn2int.x86_64
 exe/x86_64.amzn.2 prompt/off command/verifiedpermissions.batch-is-authorized-with-
token",
    "requestParameters": {
        "policyStoreId": "PSEXAMPLEabcdefg111111",
        "requests": [
```

```
            {
                "action": {
                    "actionType": "PhotoFlash::Action",
                    "actionId": "ViewPhoto"
                },
                "resource": {
                    "entityType": "PhotoFlash::Photo",
                    "entityId": "VacationPhoto94.jpg"
                }
            },
            {
                "action": {
                    "actionType": "PhotoFlash::Action",
                    "actionId": "DeletePhoto"
                },
                "resource": {
                    "entityType": "PhotoFlash::Photo",
                    "entityId": "VacationPhoto94.jpg"
                }
            }
        ]
    },
    "responseElements": null,
    "additionalEventData": {
        "principal": {
            "entityType": "PhotoFlash::User",
            "entityId": "us-east-1_EXAMPLE|EXAMPLE_SUBJECT_ID"
        },
        "results": [
            {
                "request": {
                    "action": {
                        "actionType": "PhotoFlash::Action",
                        "actionId": "ViewPhoto"
                    },
                    "resource": {
                        "entityType": "PhotoFlash::Photo",
                        "entityId": "VacationPhoto94.jpg"
                    }
                },
                "decision": "ALLOW"
            },
            {
                "request": {
```

```
                    "action": {
                        "actionType": "PhotoFlash::Action",
                        "actionId": "DeletePhoto"
                    },
                    "resource": {
                        "entityType": "PhotoFlash::Photo",
                        "entityId": "VacationPhoto94.jpg"
                    }
                },
                "decision": "DENY"
            }
        ]
    },
    "requestID": "c67aa1d9-9fd7-45ad-9a23-e8f9f6dd66c8",
    "eventID": "5ed90624-a4f7-4ef6-a07a-0e6486d30fbe",
    "readOnly": true,
    "resources": [
        {
    "accountId": "123456789012",
    "type": "AWS::VerifiedPermissions::PolicyStore",
    "ARN": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data"
}
```

CreatePolicyStore

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `CreatePolicyStore` API:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:role/ExampleRole",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
```

```
  },
  "eventTime": "2023-05-22T07:43:33Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "CreatePolicyStore",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "clientToken": "a1b2c3d4-e5f6-a1b2-c3d4-TOKEN1111111",
    "validationSettings": {
      "mode": "OFF"
    }
  },
  "responseElements": {
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111",
    "createdDate": "2023-05-22T07:43:33.962794Z",
    "lastUpdatedDate": "2023-05-22T07:43:33.962794Z"
  },
  "requestID": "1dd9360e-e2dc-4554-ab65-b46d2cf45c29",
  "eventID": "b6edaeee-3584-4b4e-a48e-311de46d7532",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789012",
  "eventCategory": "Management"
}
```

ListPolicyStores

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `ListPolicyStores` API:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:role/ExampleRole",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
```

```
  "eventTime": "2023-05-22T07:43:33Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "ListPolicyStores",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "maxResults": 10
  },
  "responseElements": null,
  "requestID": "5ef238db-9f87-4f37-ab7b-6cf0ba5df891",
  "eventID": "b0430fb0-12c3-4cca-8d05-84c37f99c51f",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789012",
  "eventCategory": "Management"
}
```


DeletePolicyStore

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `DeletePolicyStore` API:


```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:role/ExampleRole",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-22T07:43:32Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "DeletePolicyStore",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "policyStoreId": "PSEXAMPLEabcdefg111111"
  },
  "responseElements": null,
```

```
  "requestID": "1368e8f9-130d-45a5-b96d-99097ca3077f",
  "eventID": "ac482022-b2f6-4069-879a-dd509123d8d7",
  "readOnly": false,
  "resources": [
    {
      "accountId": "123456789012",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789012",
  "eventCategory": "Management"
}
```

PutSchema

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `PutSchema` API:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:role/ExampleRole",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-16T12:58:57Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "PutSchema",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "policyStoreId": "PSEXAMPLEabcdefg111111"
  },
  "responseElements": {
    "lastUpdatedDate": "2023-05-16T12:58:57.513442Z",
    "namespaces": "[some_namespace]",
```

```
    "createdDate": "2023-05-16T12:58:57.513442Z",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
  },
  "requestID": "631fbfa1-a959-4988-b9f8-f1a43ff5df0d",
  "eventID": "7cd0c677-733f-4602-bc03-248bae581fe5",
  "readOnly": false,
  "resources": [
    {
      "accountId": "123456789012",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "ARN": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789012",
  "eventCategory": "Management"
}
```

GetSchema

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `GetSchema` API:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::222222222222:role/ExampleRole",
    "accountId": "222222222222",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-25T01:12:07Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "GetSchema",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "policyStoreId": "PSEXAMPLEabcdefg111111"
  },
```

```
  "responseElements": null,
  "requestID": "a1f4d4cd-6156-480a-a9b8-e85a71dcc7c2",
  "eventID": "0b3b8e3d-155c-46f3-a303-7e9e8b5f606b",
  "readOnly": true,
  "resources": [
    {
      "accountId": "222222222222",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "ARN": "arn:aws:verifiedpermissions::222222222222:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "222222222222",
  "eventCategory": "Management"
}
```

CreatePolicyTemplate

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `CreatePolicyTemplate` API:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:role/ExampleRole",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-16T13:00:24Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "CreatePolicyTemplate",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "policyStoreId": "PSEXAMPLEabcdefg111111"
  },
  "responseElements": {
    "lastUpdatedDate": "2023-05-16T13:00:23.444404Z",
```

```
    "createdDate": "2023-05-16T13:00:23.444404Z",
    "policyTemplateId": "PTEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
  },
  "requestID": "73953bda-af5e-4854-afe2-7660b492a6d0",
  "eventID": "7425de77-ed84-4f91-a4b9-b669181cc57b",
  "readOnly": false,
  "resources": [
    {
      "accountId": "123456789012",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789012",
  "eventCategory": "Management"
}
```

DeletePolicyTemplate

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `DeletePolicyTemplate` API:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::222222222222:role/ExampleRole",
    "accountId": "222222222222",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-25T01:11:48Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "DeletePolicyTemplate",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "policyStoreId": "PSEXAMPLEabcdefg111111",
```

```
    "policyTemplateId": "PTEXAMPLEabcdefg111111"
  },
  "responseElements": null,
  "requestID": "5ff0f22e-6bbd-4b85-a400-4fb74aa05dc6",
  "eventID": "c0e0c689-369e-4e95-a9cd-8de113d47ffa",
  "readOnly": false,
  "resources": [
    {
      "accountId": "222222222222",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "ARN": "arn:aws:verifiedpermissions::222222222222:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "222222222222",
  "eventCategory": "Management"
}
```

CreatePolicy

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `CreatePolicy` API:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:role/ExampleRole",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-22T07:42:30Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "CreatePolicy",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "clientToken": "a1b2c3d4-e5f6-a1b2-c3d4-TOKEN1111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
```

```
  },
  "responseElements": {
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyId": "SPEXAMPLEabcdefg111111",
    "policyType": "STATIC",
    "principal": {
        "entityType": "PhotoApp::Role",
        "entityId": "PhotoJudge"
    },
    "resource": {
        "entityType": "PhotoApp::Application",
        "entityId": "PhotoApp"
    },
    "lastUpdatedDate": "2023-05-22T07:42:30.70852Z",
    "createdDate": "2023-05-22T07:42:30.70852Z"
  },
  "requestID": "93ffa151-3841-4960-9af6-30a7f817ef93",
  "eventID": "30ab405f-3dff-43ff-8af9-f513829e8bde",
  "readOnly": false,
  "resources": [
    {
      "accountId": "123456789012",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789012",
  "eventCategory": "Management"
}
```

GetPolicy

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `GetPolicy` API:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
```

```
    "arn": "arn:aws:iam::123456789012:role/ExampleRole",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-22T07:43:29Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "GetPolicy",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyId": "SPEXAMPLEabcdefg111111"
  },
  "responseElements": null,
  "requestID": "23022a9e-2f5c-4dac-b653-59e6987f2fac",
  "eventID": "9b4d5037-bafa-4d57-b197-f46af83fc684",
  "readOnly": true,
  "resources": [
    {
      "accountId": "123456789012",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "123456789012",
  "eventCategory": "Management"
}
```

CreateIdentitySource

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `CreateIdentitySource` API:

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::333333333333:role/ExampleRole",
```

```
    "accountId": "333333333333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-19T01:27:44Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "CreateIdentitySource",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "clientToken": "a1b2c3d4-e5f6-a1b2-c3d4-TOKEN1111111",
    "configuration": {
      "cognitoUserPoolConfiguration": {
        "userPoolArn": "arn:aws:cognito-idp:000011112222:us-east-1:userpool/us-
east-1_aaaaaaaaaa"
      }
    },
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "principalEntityType": "User"
  },
  "responseElements": {
    "createdDate": "2023-07-14T15:05:01.599534Z",
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "lastUpdatedDate": "2023-07-14T15:05:01.599534Z",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
    },
  "requestID": "afcc1e67-d5a4-4a9b-a74c-cdc2f719391c",
  "eventID": "f13a41dc-4496-4517-aeb8-a389eb379860",
  "readOnly": false,
  "resources": [
    {
      "accountId": "333333333333",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "arn": "arn:aws:verifiedpermissions::333333333333:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "333333333333",
  "eventCategory": "Management"
}
```

GetIdentitySource

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `GetIdentitySource` API:


```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::333333333333:role/ExampleRole",
    "accountId": "333333333333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-24T19:55:31Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "GetIdentitySource",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
  },
  "responseElements": null,
  "requestID": "7a6ecf79-c489-4516-bb57-9ded970279c9",
  "eventID": "fa158e6c-f705-4a15-a731-2cdb4bd9a427",
  "readOnly": true,
  "resources": [
    {
      "accountId": "333333333333",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "arn": "arn:aws:verifiedpermissions::333333333333:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "333333333333",
  "eventCategory": "Management"
}
```

ListIdentitySources

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `ListIdentitySources` API:


```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::333333333333:role/ExampleRole",
    "accountId": "333333333333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-24T20:05:32Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "ListIdentitySources",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "policyStoreId": "PSEXAMPLEabcdefg111111"
  },
  "responseElements": null,
  "requestID": "95d2a7bc-7e9a-4efe-918e-97e558aacaf7",
  "eventID": "d3dc53f6-1432-40c8-9d1d-b9eeb75c6193",
  "readOnly": true,
  "resources": [
    {
      "accountId": "333333333333",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "arn": "arn:aws:verifiedpermissions::333333333333:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "333333333333",
  "eventCategory": "Management"
}
```

DeleteIdentitySource

El siguiente ejemplo muestra una entrada de CloudTrail registro para una llamada a la `DeleteIdentitySource` API:


```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE_PRINCIPAL_ID",
    "arn": "arn:aws:iam::333333333333:role/ExampleRole",
    "accountId": "333333333333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
  },
  "eventTime": "2023-05-24T19:55:32Z",
  "eventSource": "verifiedpermissions.amazonaws.com",
  "eventName": "DeleteIdentitySource",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-rust/0.55.2 os/linux lang/rust/1.69.0",
  "requestParameters": {
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
  },
  "responseElements": null,
  "requestID": "d554d964-0957-4834-a421-c417bd293086",
  "eventID": "fe4d867c-88ee-4e5d-8d30-2fbc208c9260",
  "readOnly": false,
  "resources": [
    {
      "accountId": "333333333333",
      "type": "AWS::VerifiedPermissions::PolicyStore",
      "arn": "arn:aws:verifiedpermissions::333333333333:policy-store/
PSEXAMPLEabcdefg111111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "333333333333",
  "eventCategory": "Management"
}
```

Creación de recursos de permisos verificados de Amazon con AWS CloudFormation

Amazon Verified Permissions está integrado con AWS CloudFormation un servicio que le ayuda a modelar y configurar sus AWS recursos para que pueda dedicar menos tiempo a crear y administrar sus recursos e infraestructura. Usted crea una plantilla que describe todos los AWS recursos que desea (como los almacenes de políticas) y CloudFormation aprovisiona y configura esos recursos por usted.

Cuando la utilice CloudFormation, podrá reutilizar la plantilla para configurar los recursos de permisos verificados de forma coherente y repetida. Describa sus recursos una vez y, a continuación, aprovisione los mismos recursos una y otra vez en varias Cuentas de AWS regiones.

> **Important**
>
> Amazon Cognito Identity no está disponible al mismo tiempo que los permisos verificados de Regiones de AWS Amazon. Si recibe un error CloudFormation relacionado con Amazon Cognito Identity, por ejemplo, le recomendamos que cree el grupo de Amazon Cognito usuarios y el cliente en la zona geográfica más cercana Región de AWS a donde esté disponible Amazon Cognito Identity. `Unrecognized resource types: AWS::Cognito::UserPool, AWS::Cognito::UserPoolClient` Utilice este grupo de usuarios recién creado al crear la fuente de identidad de Verified Permissions.

Permisos y plantillas verificados CloudFormation

Para aprovisionar y configurar los recursos de Verified Permissions y sus servicios relacionados, debe entender las plantillas de CloudFormation. Las plantillas son archivos de texto con formato JSON o YAML. Estas plantillas describen los recursos que deseas aprovisionar en tus CloudFormation pilas. Si no estás familiarizado con JSON o YAML, puedes usar CloudFormation Designer para ayudarte a empezar con CloudFormation las plantillas. Para obtener más información, consulta ¿Qué es CloudFormation Designer? en la Guía AWS CloudFormation del usuario.

Verified Permissions permite crear fuentes de identidad, políticas, almacenes de políticas, plantillas de políticas y alias de almacenes de políticas en CloudFormation. Para obtener más información, incluidos ejemplos de plantillas JSON y YAML para los recursos de Verified Permissions, consulte

la referencia del tipo de recurso de Amazon Verified Permissions en la Guía el usuario de AWS CloudFormation .

AWS CDK construye

Kit de desarrollo de la nube de AWS (AWS CDK) Se trata de un marco de desarrollo de software de código abierto para definir la infraestructura de nube en el código y aprovisionarla mediante ella. CloudFormation Se pueden usar construcciones, o componentes de nube reutilizables, para crear plantillas. CloudFormation Luego, estas plantillas se pueden usar para implementar su infraestructura de nube.

Para obtener más información y descargar el AWS CDK, consulte AWS Cloud Development Kit.

Los siguientes son enlaces a la documentación sobre los AWS CDK recursos de permisos verificados, como las construcciones.

- Amazon Verified Permissions L2 CDK Construct

Más información sobre CloudFormation

Para obtener más información CloudFormation, consulte los siguientes recursos:

- AWS CloudFormation
- AWS CloudFormation Guía del usuario
- CloudFormation Referencia de la API
- AWS CloudFormation Guía del usuario de la interfaz de línea de comandos

Acceda a los permisos verificados de Amazon mediante AWS PrivateLink

Puedes usarlo AWS PrivateLink para crear una conexión privada entre tu VPC y los permisos verificados de Amazon. Puede acceder a los permisos verificados como si estuvieran en su VPC, sin usar una puerta de enlace a Internet, un dispositivo NAT, una conexión VPN o Direct Connect una conexión. Las instancias de la VPC no necesitan direcciones IP públicas para acceder a Verified Permissions.

Esta conexión privada se establece mediante la creación de un punto de conexión de interfaz alimentado por AWS PrivateLink. Creamos una interfaz de red de punto de conexión en cada subred habilitada para el punto de conexión de interfaz. Se trata de interfaces de red administradas por el solicitante que sirven como punto de entrada para el tráfico destinado a Verified Permissions.

Para obtener más información, consulte Acceso a los Servicios de AWS a través de AWS PrivateLink en la Guía de AWS PrivateLink .

Consideraciones sobre Verified Permissions

Antes de configurar un punto de conexión de interfaz para Verified Permissions, consulte la sección Consideraciones en la Guía de AWS PrivateLink .

Verified Permissions admite la realización de llamadas a todas las acciones de la API a través del punto de conexión de interfaz.

Las políticas de punto de conexión de VPC son compatibles con Verified Permissions. De forma predeterminada, el acceso completo a Verified Permissions se permite a través del punto de conexión. Como alternativa, puede asociar un grupo de seguridad a las interfaces de red de los puntos de conexión para controlar el tráfico a Verified Permissions a través del punto de conexión de interfaz.

Crear un punto de conexión de interfaz para Verified Permissions

Puede crear un punto de conexión de interfaz para Verified Permissions mediante la consola de Amazon VPC o la AWS Command Line Interface (AWS CLI). Para obtener más información, consulte Creación de un punto de conexión de interfaz en la Guía de AWS PrivateLink .

Cree un punto de conexión para Verified Permissions utilizando el siguiente nombre de servicio:

```
com.amazonaws.region.verifiedpermissions
```

Si habilita DNS privado para el punto de conexión de interfaz, puede realizar solicitudes a la API para Verified Permissions usando su nombre de DNS predeterminado para la región. Por ejemplo, `verifiedpermissions.us-east-1.amazonaws.com`.

Creación de una política de puntos de conexión para el punto de conexión de interfaz

Una política de punto de conexión es un recurso de IAM que puede adjuntar al punto de conexión de su interfaz. La política de puntos finales predeterminada permite el acceso total a los permisos verificados a través del punto final de la interfaz. Para controlar el acceso permitido a los permisos verificados desde su VPC, adjunte una política de punto final personalizada al punto final de la interfaz.

Una política de punto de conexión especifica la siguiente información:

- Las entidades principales que pueden llevar a cabo acciones (Cuentas de AWS, usuarios de IAM y roles de IAM).
- Las acciones que se pueden realizar.
- El recurso en el que se pueden realizar las acciones.

Para obtener más información, consulte Control del acceso a los servicios con políticas de punto de conexión en la Guía del usuario de AWS PrivateLink .

Ejemplo: política de puntos finales de VPC para acciones de permisos verificados

El siguiente es un ejemplo de una política de un punto de conexión personalizado. Al adjuntar esta política al punto final de la interfaz, concede acceso a las acciones de permisos verificados enumeradas a todos los directores de todos los recursos.

```
{
   "Statement": [
      {
         "Principal": "*",
         "Effect": "Allow",
```

```
         "Action": [
            "verifiedpermissions:IsAuthorized",
            "verifiedpermissions:IsAuthorizedWithToken",
            "verifiedpermissions:GetPolicy"
         ],
         "Resource":"*"
      }
   ]
}
```

Cuotas para Amazon Verified Permissions

¿Tu Cuenta de AWS tiene cuotas predeterminadas, antes denominadas límites, para cada AWS servicio. A menos que se indique lo contrario, cada cuota es Region-specific. Puede solicitar el aumento de algunas cuotas, pero otras no se pueden aumentar.

Para ver todas las cuotas de Verified Permissions, abra la consola de Service Quotas. En el panel de navegación, elija Servicios de AWS y seleccione Verified Permissions.

Para solicitar un aumento de cuota, consulte Solicitud de aumento de cuota en la Guía del usuario de Service Quotas. Si la cuota aún no se encuentra disponible en Service Quotas, utilice el formulario de aumento del límite.

Tu Cuenta de AWS tiene las siguientes cuotas relacionadas con los permisos verificados.

Temas

- Cuotas de recursos
- Cuotas para jerarquías
- Cuotas de operaciones por segundo

Cuotas de recursos

Name	Predeterminado	Ajusta	Description (Descripción)
Almacenes de políticas por región y por cuenta	Cada región admitida: 30 000	Sí	El número máximo de almacenes de políticas.
Plantillas de política por almacén de políticas	Cada región admitida: 40	Sí	El número máximo de plantillas de política en un almacén de políticas.
Fuentes de identidad por almacén de políticas	1	No	El número máximo de fuentes de identidad que puede definir para un almacén de políticas.

Name	Predeterminado	Ajusta	Description (Descripción)
Alias de almacén de políticas por almacén de políticas	10	Sí	El número máximo de alias de almacén de políticas que puede asociar a un único almacén de políticas.
Tamaño de la solicitud de autorización[1]	1 MB	No	Tamaño máximo de una solicitud de autorización.
Tamaño de la política	10 000 bytes	Sí	El tamaño máximo de una política individual.
Tamaño del esquema	100 000 bytes	Sí	El tamaño máximo del esquema de un almacén de políticas.
Espacios de nombres por esquema de almacén de políticas	100	Sí	La cantidad máxima de espacios de nombres que puede definir en un esquema de almacén de políticas.
Tamaño de la política por recurso	1 MB[2]	Sí	El tamaño máximo de todas las políticas que hacen referencia a un recurso específico.
Tamaño de la póliza por combinación principal de recursos	200 000 bytes[2]	No	El tamaño máximo de todas las pólizas que hacen referencia a una combinación específic a de un principal y un recurso.

[1] La cuota para una solicitud de autorización es la misma para ambas IsAuthorized y IsAuthorizedWithToken.

[2] El tamaño total de todas las políticas para una combinación única de un principal y un recurso no puede superar los 200 000 bytes. Un principal o recurso no especificado cuenta como un valor único; por ejemplo, el límite se aplica al tamaño total de todas las políticas asignadas a un principal en particular sin especificar el recurso o a un recurso en particular sin especificar. Este límite no se puede ajustar. Además, el tamaño total de todas las políticas para un recurso único (o un recurso no especificado) no debe superar 1 MB; este límite se puede ajustar con restricciones adicionales en las cuotas jerárquicas. Para obtener más información sobre cómo Verified Permissions calcula esta cuota, así como ejemplos y estrategias para mantenerse dentro de ella, consulte Tamaño de la política por recurso.

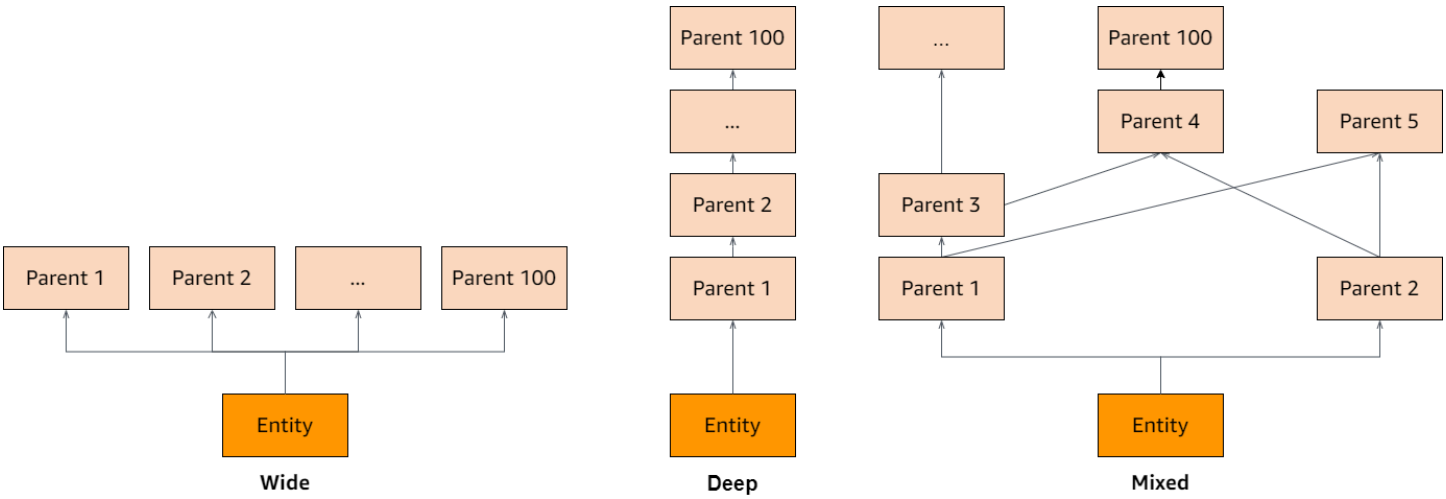
Cuotas para jerarquías

> **Note**
>
> Cada cuota de padres transitivos se aplica a cada entidad de forma individual y cuenta tanto a los padres directos como a los indirectos (transitivos) en el total. Por ejemplo, si el límite de Padres transitivos por director es 100, un director puede tener 100 grupos de padres directos sin grupos anidados. Como alternativa, un principal puede tener 10 grupos de padres directos, cada uno de los cuales pertenezca a 9 grupos de padres adicionales. Cualquier combinación que sume un total de 100 padres para ese director es válida.

Name	Predeterminado	Ajusta	Description (Descripción)
Elementos principales transitivos por entidad principal	100	No	El número máximo de elementos principales transitivos para cada entidad principal.
Elementos principales transitivos por acción	100	No	El número máximo de elementos principales transitivos para cada acción.
Elementos principales transitivos por recurso	100	No	El número máximo de elementos principales

Name	Predeterminado	Ajusta	Description (Descripción)
			transitivos para cada recurso.

El siguiente diagrama ilustra cómo se pueden definir los elementos principales transitivos para una entidad (principal, acción o recurso).



Cuotas de operaciones por segundo

Los permisos verificados limitan las solicitudes a los puntos finales de servicio en un Región de AWS cuando las solicitudes de aplicaciones superan la cuota de una operación de API. Los permisos verificados pueden generar una excepción si superas la cuota de solicitudes por segundo o si intentas realizar operaciones de escritura simultáneas. Puedes ver tus cuotas de RPS actuales en Cuotas de servicio. Para evitar que las aplicaciones superen la cuota de una operación, debe optimizarlas para que tengan en cuenta los reintentos y los retrasos exponenciales. Para obtener más información, consulte Vuelva a intentarlo con un patrón de retroceso y Administrar y supervisar la limitación de las API en sus cargas de trabajo.

Name	Predeterminado	Ajusta	Description (Descripción)
BatchGetPolicy solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de BatchGetPolicy solicitudes por segundo por almacén de políticas.

Name	Predeterminado	Ajusta	Description (Descripción)
BatchIsAuthorized solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 30	Sí	El número máximo de BatchIsAuthorized solicitudes por segundo por almacén de políticas.
BatchIsAuthorizedWithToken solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 30	Sí	El número máximo de BatchIsAuthorizedWithToken solicitudes por segundo por almacén de políticas.
CreateIdentitySource solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 1	Sí	El número máximo de CreateIdentitySource solicitudes por segundo por almacén de políticas.
CreatePolicy solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de CreatePolicy solicitudes por segundo por almacén de políticas.
CreatePolicyStore solicitudes por segundo, por región y por cuenta	Cada región admitida: 1	No	El número máximo de CreatePolicyStore solicitudes por segundo.
CreatePolicyTemplate solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de CreatePolicyTemplate solicitudes por segundo por almacén de políticas.
DeleteIdentitySource solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 1	Sí	El número máximo de DeleteIdentitySource solicitudes por segundo por almacén de políticas.

Name	Predeterminado	Ajusta	Description (Descripción)
DeletePolicy solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de DeletePolicy solicitudes por segundo por almacén de políticas.
DeletePolicyStore solicitudes por segundo, por región y por cuenta	Cada región admitida: 1	No	El número máximo de DeletePolicyStore solicitudes por segundo.
DeletePolicyTemplate solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de DeletePolicyTemplate solicitudes por segundo por almacén de políticas.
GetIdentitySource solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de GetIdentitySource solicitudes por segundo por almacén de políticas.
GetPolicy solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de GetPolicy solicitudes por segundo por almacén de políticas.
GetPolicyStore solicitudes por segundo, por región y por cuenta	Cada región admitida: 10	Sí	El número máximo de GetPolicyStore solicitudes por segundo.
GetPolicyTemplate solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de GetPolicyTemplate solicitudes por segundo por almacén de políticas.

Name	Predeterminado	Ajusta	Description (Descripción)
GetSchema solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de GetSchema solicitudes por segundo por almacén de políticas.
IsAuthorized solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 200	Sí	El número máximo de IsAuthorized solicitudes por segundo por almacén de políticas.
IsAuthorizedWithToken solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 200	Sí	El número máximo de IsAuthorizedWithToken solicitudes por segundo por almacén de políticas.
ListIdentitySources solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de ListIdentitySources solicitudes por segundo por almacén de políticas.
ListPolicies solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de ListPolicies solicitudes por segundo por almacén de políticas.
ListPolicyStores solicitudes por segundo, por región y por cuenta	Cada región admitida: 10	Sí	El número máximo de ListPolicyStores solicitudes por segundo.
ListPolicyTemplates solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de ListPolicyTemplates solicitudes por segundo por almacén de políticas.

Name	Predeterminado	Ajusta	Description (Descripción)
PutSchema solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de PutSchema solicitudes por segundo por almacén de políticas.
UpdateIdentitySource solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 1	Sí	El número máximo de UpdateIdentitySource solicitudes por segundo por almacén de políticas.
UpdatePolicy solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de UpdatePolicy solicitudes por segundo por almacén de políticas.
UpdatePolicyStore solicitudes por segundo, por región y por cuenta	Cada región admitida: 10	No	El número máximo de UpdatePolicyStore solicitudes por segundo.
UpdatePolicyTemplate solicitudes por segundo, por región, por almacén de políticas	Cada región admitida: 10	Sí	El número máximo de UpdatePolicyTemplate solicitudes por segundo por almacén de políticas.

Términos y conceptos lingüísticos de la política de permisos verificados de Amazon y Cedar

Debe comprender los siguientes conceptos para utilizar Amazon Verified Permissions.

Conceptos de Verified Permissions

- Modelo de autorización
- Solicitud de autorización
- Respuesta de autorización
- Políticas consideradas
- Datos de contexto
- Políticas determinantes
- Datos de la entidad
- Permisos, autorizaciones y entidades principales
- Aplicación de políticas
- Almacén de políticas
- Alias del almacén de políticas
- Nombre de la política
- Nombre de la plantilla de política
- Políticas satisfechas
- Diferencias entre los permisos verificados de Amazon y el lenguaje de la política de Cedar

Conceptos del lenguaje de políticas de Cedar

- Autorización
- Entidad
- Grupos y jerarquías
- Espacios de nombres
- Política
- Plantilla de política
- Esquema

Modelo de autorización

El modelo de autorización describe el alcance de las solicitudes de autorización realizadas por la aplicación y la base para evaluarlas. Se define en términos de los diferentes tipos de recursos, las acciones que se realizan sobre esos recursos y los tipos de entidad principal que toman esas acciones. También considera el contexto en el que se llevan a cabo esas acciones.

El control de acceso basado en roles (RBAC) es una base de evaluación en la que los roles se definen y asocian a un conjunto de permisos. Después, estas roles se pueden asignar a una o más identidades. La identidad asignada adquiere los permisos asociados al rol. Si se modifican los permisos asociados al rol, la modificación afecta automáticamente a cualquier identidad a la que se haya asignado el rol. Cedar puede respaldar las decisiones del RBAC mediante el uso de grupos de entidades principales.

El control de acceso basado en atributos (ABAC) es una base de evaluación en la que los permisos asociados a una identidad vienen determinados por los atributos de esa identidad. Cedar puede respaldar las decisiones del ABAC mediante el uso de condiciones políticas que hagan referencia a los atributos de la entidad principal.

El lenguaje de políticas de Cedar permite combinar RBAC y ABAC en una sola política al permitir definir los permisos para un grupo de usuarios, que tienen condiciones basadas en atributos.

Solicitud de autorización

Una solicitud de autorización es una solicitud de Verified Permissions realizada por una aplicación para evaluar un conjunto de políticas a fin de determinar si una entidad principal puede realizar una acción en un recurso en un contexto determinado.

Respuesta de autorización

La respuesta de autorización es la respuesta a la solicitud de autorización. Incluye una decisión de permitir o denegar, además de información adicional, como IDs las políticas determinantes.

Políticas consideradas

Las políticas consideradas son el conjunto completo de políticas que Verified Permissions selecciona para incluirlas al evaluar una solicitud de autorización.

Datos de contexto

Los datos de contexto son valores de atributos que proporcionan información adicional para su evaluación.

Políticas determinantes

Las políticas determinantes son las políticas que determinan la respuesta de autorización. Por ejemplo, si hay dos políticas satisfechas, una de denegación y la otra de permiso, la política de denegación será la política determinante. Si hay varias políticas de permisos satisfechas y ninguna política de prohibición satisfecha, hay varias políticas determinantes. En el caso de que ninguna política coincida y la respuesta sea denegar, no hay políticas determinantes.

Datos de la entidad

Los datos de la entidad son los datos sobre la entidad principal, la acción y el recurso. Los datos de la entidad relevantes para la evaluación de las políticas son la pertenencia a grupos hasta el final de la jerarquía de la entidad y los valores de los atributos de la entidad principal y el recurso.

Permisos, autorizaciones y entidades principales

Verified Permissions administra los permisos y la autorización detallados en las aplicaciones personalizadas que usted crea.

Una entidad principal es el usuario de una aplicación, ya sea humana o automática, que tiene una identidad vinculada a un identificador, como un nombre de usuario o un identificador de máquina. El proceso de autenticación determina si la entidad principal es realmente la identidad que afirma ser.

Asociado a esa identidad hay un conjunto de permisos de aplicación que determinan lo que la entidad principal puede hacer dentro de esa aplicación. La autorización es el proceso de evaluar esos permisos para determinar si una entidad principal está autorizada a realizar una acción determinada en la aplicación. Estos permisos se pueden expresar como políticas.

Aplicación de políticas

La aplicación de las políticas es el proceso de hacer cumplir la decisión de evaluación dentro de la aplicación fuera de Verified Permissions. Si la evaluación de Verified Permissions da como resultado una denegación, la aplicación garantizaría que se impidiera a la entidad principal acceder al recurso.

Almacén de políticas

Un almacén de políticas es un contenedor de políticas y plantillas. Cada almacén contiene un esquema que se utiliza para validar las políticas agregadas al almacén. De forma predeterminada, cada aplicación tiene su propio almacén de políticas, pero varias aplicaciones pueden compartir un único almacén de políticas. Cuando una aplicación realiza una solicitud de autorización, identifica el almacén de políticas utilizado para evaluar esa solicitud. Los almacenes de políticas proporcionan una forma de aislar un conjunto de políticas y, por lo tanto, se pueden usar en una aplicación de varios inquilinos para incluir los esquemas y las políticas de cada inquilino. Una sola aplicación puede tener almacenes de políticas independientes para cada inquilino.

Al evaluar una solicitud de autorización, Verified Permissions solo tiene en cuenta el subconjunto de las políticas del almacén de políticas que son relevantes para la solicitud. La relevancia se determina en función del alcance de la política. El alcance identifica la entidad principal y el recurso específicos a los que se aplica la política, así como las acciones que la entidad principal puede realizar en el recurso. Definir el alcance ayuda a mejorar el rendimiento al reducir el conjunto de políticas consideradas.

Alias del almacén de políticas

Un alias de almacén de políticas es un nombre descriptivo para un almacén de políticas. Puede usar un alias de almacén de políticas para identificar un almacén de políticas en cualquier operación de permisos verificados que acepte un `policyStoreId` parámetro. Los alias de los almacenes de políticas son AWS recursos independientes con sus propios ARNs recursos. Cada alias se asocia a un almacén de políticas a la vez y se pueden asociar varios alias al mismo almacén de políticas. Para obtener más información, consulte Alias del almacén de políticas de permisos verificados de Amazon.

Nombre de la política

El nombre de una política es un nombre descriptivo opcional para una política. Los nombres de las políticas deben ser únicos para todas las políticas del almacén de políticas y tener el prefijo. `name/` Puede utilizar un nombre de política en lugar del identificador de la política en las operaciones del plano de control que acepten un `policyId` parámetro. Los nombres se pueden establecer al crear o actualizar una política. Solo `GetPolicy` `ListPolicies` devuelve el nombre en la salida.

Nombre de la plantilla de política

El nombre de una plantilla de política es un nombre descriptivo opcional para una plantilla de política. Los nombres de las plantillas de políticas deben ser únicos para todas las plantillas de políticas del almacén de políticas y tener el prefijo `dename/`. Puede utilizar un nombre de plantilla de política en lugar del ID de la plantilla de política en las operaciones del plano de control que aceptan un `policyTemplateId` parámetro. Los nombres se pueden establecer al crear o actualizar una plantilla de política. Solo `GetPolicyTemplate` `ListPolicyTemplates` devuelve el nombre en la salida.

Políticas satisfechas

Las políticas satisfechas son las políticas que coinciden con los parámetros de la solicitud de autorización.

Diferencias entre los permisos verificados de Amazon y el lenguaje de la política de Cedar

Amazon Verified Permissions utiliza el motor de lenguaje de políticas de Cedar para realizar las tareas de autorización. Sin embargo, existen algunas diferencias entre la implementación nativa de Cedar y la implementación de Cedar en Verified Permissions. En este tema se explican esas diferencias.

> **Note**
>
> A partir de mayo de 2026, Verified Permissions se alineó con Cedar y ahora admite varios espacios de nombres.

Compatibilidad con las plantillas de política

Tanto Verified Permissions como Cedar solo permiten marcadores de posición en el ámbito para `principal` y `resource`. Sin embargo, Verified Permissions también requiere que ni `principal` ni `resource` carezcan de restricciones.

La siguiente política es válida en Cedar, pero Verified Permissions la rechaza porque la `principal` no tiene restricciones.

```
permit(principal, action == Action::"view", resource == ?resource);
```

Los dos ejemplos siguientes son válidos tanto en Cedar como en Verified Permissions porque la `principal` y el `resource` tienen restricciones.

```
permit(principal == User::"alice", action == Action::"view", resource == ?resource);
```

```
permit(principal == ?principal, action == Action::"a", resource in ?resource);
```

Compatibilidad con esquemas

Verified Permissions requiere que todos los nombres de claves JSON del esquema no sean cadenas vacías. Cedar permite cadenas vacías en algunos casos, como en el caso de propiedades o espacios de nombres.

Definición de grupos de acción

Los métodos de autorización de Cedar requieren una lista de las entidades que se deben tener en cuenta al evaluar una solicitud de autorización con respecto a las políticas.

Puede definir las acciones y los grupos de acciones que utiliza su aplicación en el esquema. Sin embargo, Cedar no incluye el esquema como parte de una solicitud de evaluación. En su lugar, Cedar usa el esquema solo para validar las políticas y las plantillas de políticas que envíe. Dado que Cedar no hace referencia al esquema durante las solicitudes de evaluación, aunque haya definido grupos de acciones en el esquema, debe incluir también la lista de todos los grupos de acciones como parte de la lista de entidades que debe transferir a las operaciones de la API de autorización.

Verified Permissions lo hace por usted. Todos los grupos de acciones que defina en su esquema se anexan automáticamente a la lista de entidades que pase como parámetro de las operaciones `IsAuthorized` o `IsAuthorizedWithToken`.

Formato de entidades

El formato JSON de las entidades en los permisos verificados que utilizan el `entityList` parámetro difiere del de Cedar en los siguientes aspectos:

- En Verified Permissions, un objeto JSON debe tener todos sus pares clave-valor incluidos en un objeto JSON con el nombre de `Record`.

- Una lista JSON de Verified Permissions debe estar encapsulada en un par clave-valor de JSON en el que el nombre de la clave sea `Set` y el valor sea la lista JSON original de Cedar.
- En el caso de los nombres de tipo `String`, `Long` y `Boolean`, cada par clave-valor de Cedar se sustituye por un objeto JSON en Verified Permissions. El nombre del objeto es el nombre de la clave original. Dentro del objeto JSON, hay un par clave-valor en el que el nombre de la clave es el nombre de tipo del valor escalar (`String`, `Long` o `Boolean`) y el valor es el de la entidad de Cedar.
- El formato de la sintaxis de las entidades de Cedar y Verified Permissions difiere en los siguientes aspectos:

Formato de Cedar	Formato de Verified Permissions
`uid`	`Identifier`
`type`	`EntityType`
`id`	`EntityId`
`attrs`	`Attributes`
`parents`	`Parents`

Example- Listas

Los siguientes ejemplos muestran cómo se expresa una lista de entidades en Cedar y Verified Permissions, respectivamente.

Cedar

```
[
  {
    "number": 1
  },
  {
    "sentence": "Here is an example sentence"
  },
  {
    "Question": false
  }
```

```
]
```

Verified Permissions

```
{
  "Set": [
    {
      "Record": {
        "number": {
          "Long": 1
        }
      }
    },
    {
      "Record": {
        "sentence": {
          "String": "Here is an example sentence"
        }
      }
    },
    {
      "Record": {
        "question": {
          "Boolean": false
        }
      }
    }
  ]
}
```

Example- Evaluación de políticas

Los siguientes ejemplos muestran cómo se formatean las entidades para evaluar una política en una solicitud de autorización en Cedar y Verified Permissions, respectivamente.

Cedar

```
[
    {
        "uid": {
            "type": "PhotoApp::User",
            "id": "alice"
```

```
        },
        "attrs": {
            "age": 25,
            "name": "alice",
            "userId": "123456789012"
        },
        "parents": [
            {
                "type": "PhotoApp::UserGroup",
                "id": "alice_friends"
            },
            {
                "type": "PhotoApp::UserGroup",
                "id": "AVTeam"
            }
        ]
    },
    {
        "uid": {
            "type": "PhotoApp::Photo",
            "id": "vacationPhoto.jpg"
        },
        "attrs": {
            "private": false,
            "account": {
                "__entity": {
                    "type": "PhotoApp::Account",
                    "id": "ahmad"
                }
            }
        },
        "parents": []
    },
    {
        "uid": {
            "type": "PhotoApp::UserGroup",
            "id": "alice_friends"
        },
        "attrs": {},
        "parents": []
    },
    {
        "uid": {
            "type": "PhotoApp::UserGroup",
```

```
            "id": "AVTeam"
        },
        "attrs": {},
        "parents": []
    }
]
```

Verified Permissions

```
[
    {
        "Identifier": {
            "EntityType": "PhotoApp::User",
            "EntityId": "alice"
        },
        "Attributes": {
            "age": {
                "Long": 25
            },
            "name": {
                "String": "alice"
            },
            "userId": {
                "String": "123456789012"
            }
        },
        "Parents": [
            {
                "EntityType": "PhotoApp::UserGroup",
                "EntityId": "alice_friends"
            },
            {
                "EntityType": "PhotoApp::UserGroup",
                "EntityId": "AVTeam"
            }
        ]
    },
    {
        "Identifier": {
            "EntityType": "PhotoApp::Photo",
            "EntityId": "vacationPhoto.jpg"
        },
        "Attributes": {
```

```
            "private": {
                "Boolean": false
            },
            "account": {
                "EntityIdentifier": {
                    "EntityType": "PhotoApp::Account",
                    "EntityId": "ahmad"
                }
            }
        },
        "Parents": []
    },
    {
        "Identifier": {
            "EntityType": "PhotoApp::UserGroup",
            "EntityId": "alice_friends"
        },
        "Parents": []
    },
    {
        "Identifier": {
            "EntityType": "PhotoApp::UserGroup",
            "EntityId": "AVTeam"
        },
        "Parents": []
    }
]
```

Límites de longitud y tamaño

Verified Permissions admite el almacenamiento en forma de almacenes de políticas para almacenar esquemas, políticas y plantillas de políticas. Ese almacenamiento hace que Verified Permissions imponga algunos límites de longitud y tamaño que no son relevantes para Cedar.

Objeto	Límite de Verified Permissions (en bytes)	Límite de Cedar
Tamaño de la política[1]	10 000	Ninguno

Objeto	Límite de Verified Permissions (en bytes)	Límite de Cedar
Descripción de la política insertada	150	No aplicable a Cedar
Tamaño de la plantilla de política	10 000	Ninguno
Tamaño del esquema	100 000	Ninguno
Tipo de identidad	200	Ninguno
ID de política	64	Ninguno
ID de plantilla de política	64	Ninguno
ID de la identidad	200	Ninguno
ID del almacén de políticas	64	No aplicable a Cedar

[1] Existe un límite de políticas por almacén de políticas en Verified Permissions en función del tamaño combinado de las entidades principales, las acciones y los recursos de las políticas creadas en el almacén de políticas. El tamaño total de todas las políticas pertenecientes a un único recurso no puede superar los 200 000 bytes. En el caso de las políticas vinculadas a plantillas, el tamaño de la plantilla de política se cuenta solo una vez, más el tamaño de cada conjunto de parámetros utilizado para crear una instancia de cada política vinculada a una plantilla.

Preguntas frecuentes sobre la actualización de Amazon Verified Permissions a Cedar 4

Amazon Verified Permissions está actualizando la versión de Cedar que utiliza de la versión 2 a la versión 4. Cedar es el lenguaje de código abierto que utiliza para escribir las políticas, las plantillas de políticas y los esquemas de sus almacenes de pólizas. Con la compatibilidad con Cedar 4 en Verified Permissions, puede utilizar nuevas funciones, como las etiquetas de `is` operador y entidad, para redactar políticas más expresivas.

Amazon Verified Permissions actualiza automáticamente los almacenes de políticas a Cedar 4. Sin embargo, algunas políticas, esquemas y solicitudes de autorización redactadas para Cedar 2 son incompatibles con Cedar 4. Si este es el caso de su almacén de pólizas, no lo actualizaremos automáticamente. Es posible que deba realizar cambios en sus políticas, plantillas de políticas, esquemas o código de aplicación antes de poder actualizar a Cedar 4.

Temas

- ¿Por qué recibo un mensaje de error que dice que Cedar 2 ya no es compatible cuando llamo a Amazon Verified Permissions?
- ¿Por qué algunas políticas, plantillas de políticas y esquemas no son compatibles con Cedar 4?
- ¿Cómo puedo saber si mi almacén de pólizas usa Cedar 2 o Cedar 4?
- ¿Cómo puedo actualizar a Cedar 4?
- ¿Puedo cambiar mi almacén de pólizas de Cedar 4 a Cedar 2?
- ¿Por qué recibo un mensaje de error que indica que mi almacén de políticas está configurado para Cedar 2?
- ¿Cómo hago que mi esquema sea compatible con Cedar 4?
- ¿Cómo puedo hacer que mis políticas y plantillas sean compatibles con Cedar 4?

¿Por qué recibo un mensaje de error que dice que Cedar 2 ya no es compatible cuando llamo a Amazon Verified Permissions?

A partir de abril de 2026, las API de autorización (`IsAuthorized`, `BatchIsAuthorized`, `IsAuthorizedWithToken` y `BatchIsAuthorizedWithToken`) se inhabilitarán en los almacenes de políticas que usen Cedar 2. En este caso, aparece el siguiente mensaje de error:

```
Your policy store uses Cedar 2.x, which is no longer supported. See https://
docs.aws.amazon.com/verifiedpermissions/latest/userguide/cedar4-faq.html or contact AWS
 Support for more information.
```

Si aparece este mensaje de error al llamar a Verified Permissions, significa que su almacén de políticas no es compatible con Cedar 4 y, por lo tanto, no se ha podido migrar a la nueva versión. Para seguir utilizando los permisos verificados de Amazon, crea un nuevo almacén de políticas, que utilizará Cedar 4, o ponte en contacto con nosotros Soporte.

¿Por qué algunas políticas, plantillas de políticas y esquemas no son compatibles con Cedar 4?

El equipo de Cedar ha realizado varios cambios incompatibles con versiones anteriores desde Cedar 2 para corregir errores y simplificar el lenguaje. Entre estos cambios se incluyen los siguientes:

- cambios en la sintaxis de las políticas, las plantillas de políticas y los esquemas
- un validador de políticas más preciso, que detecta más errores
- cambios en el comportamiento de las funciones integradas, como `isInRange`

Para obtener una lista completa de los cambios incompatibles con versiones anteriores, busca los elementos marcados (*) en el registro de cambios de Cedar.

¿Cómo puedo saber si mi almacén de pólizas usa Cedar 2 o Cedar 4?

Puedes comprobar la versión de Cedar que utiliza tu almacén de políticas mediante la consola de permisos verificados de Amazon o mediante la GetPolicyStore operación.

> **Note**
>
> Todos los almacenes de pólizas de Cuenta de AWS la misma región utilizan la misma versión de Cedar.

Console

Para comprobar la versión de Cedar de un almacén de pólizas (consola)

1. Inicia sesión en la consola de permisos verificados de Amazon Consola de administración de AWS y ábrela en https://console.aws.amazon.com/verifiedpermissions/.
2. En el panel de navegación, selecciona Almacenes de políticas y, a continuación, elige el almacén de políticas que deseas consultar.
3. Seleccione Configuración en el panel de navegación.
4. En el cuadro Detalles, busque el campo Versión de Cedar.

El campo indica `CEDAR_2` si su almacén de pólizas usa Cedar 2 y `CEDAR_4` si usa Cedar 4.

CLI

Para comprobar la versión de Cedar de un almacén de pólizas (AWS CLI)

1. Instale y configure AWS Command Line Interface (AWS CLI), si aún no lo ha hecho. Para obtener más información, consulte Instalación o actualización de la última versión de la AWS CLI.
2. Utilice el comando `get-policy-store`. En el siguiente ejemplo, *`policy-store-id`* sustitúyalo por el identificador de tu almacén de políticas:

```
aws verifiedpermissions get-policy-store \
  --policy-store-id policy-store-id
```

El `cedarVersion` campo de la salida muestra qué versión de Cedar utiliza el almacén de políticas. Por ejemplo:

```
{
    "policyStoreId": "ABCDEFG12345678abcdefg",
    "arn": "arn:aws:verifiedpermissions::111122223333:policy-store/
ABCDEFG12345678abcdefg",
    "validationSettings": {
        "mode": "STRICT"
    },
    "createdDate": "2025-06-03T13:09:47.752255+00:00",
    "lastUpdatedDate": "2025-06-03T13:09:47.752255+00:00",
    "deletionProtection": "ENABLED",
```

```
    "cedarVersion": "CEDAR_2"
}
```

El campo indica `CEDAR_2` si su almacén de políticas usa Cedar 2 y `CEDAR_4` si usa Cedar 4.

¿Cómo puedo actualizar a Cedar 4?

Amazon Verified Permissions ya ha actualizado a la mayoría de los clientes a Cedar 4. Si nunca ha creado un almacén de políticas, todos los nuevos almacenes de políticas que cree utilizarán Cedar 4. Si ya eres cliente, es probable que ya te hayamos actualizado a Cedar 4. Consulte ¿Cómo puedo saber si mi almacén de pólizas usa Cedar 2 o Cedar 4? para comprobar qué versión de Cedar utilizan sus pólizas.

Si no se ha actualizado, Verified Permissions ha detectado una política, plantilla de política, esquema o solicitud de autorización en uno de sus almacenes de políticas que no es compatible con Cedar 4. Le enviaremos una notificación por correo electrónico describiendo qué recursos son incompatibles más adelante en 2025. Para actualizar antes, abre una caja con Soporte.

> **Important**
>
> Todos los almacenes de pólizas del mismo Cuenta de AWS modo utilizan la misma versión de Cedar. Si un almacén de políticas de su cuenta no es compatible con Cedar 4, no podrá usar Cedar 4 en ningún almacén de políticas de esa cuenta.

¿Puedo cambiar mi almacén de pólizas de Cedar 4 a Cedar 2?

No. Si tiene problemas después de actualizar su almacén de pólizas a Cedar 4, abra un caso con Soporte.

¿Por qué recibo un mensaje de error que indica que mi almacén de políticas está configurado para Cedar 2?

Algunas funciones de Amazon Verified Permissions se basan en las nuevas funciones de Cedar 4. Si su almacén de políticas no usa Cedar 4, no podrá usar los siguientes campos de API:

- En las BatchIsAuthorizedWithToken operaciones IsAuthorizedBatchIsAuthorized, IsAuthorizedWithToken y:
  - `datetime`, `decimal` o `duration` valores en los `context` campos `attributes` o

No puede usar la sintaxis ni los tipos de datos en las políticas, plantillas de políticas o esquemas introducidos después de Cedar 2 hasta que se actualice su almacén de políticas.

¿Cómo hago que mi esquema sea compatible con Cedar 4?

La consola de permisos verificados puede solucionar automáticamente algunos problemas de compatibilidad en su esquema. Si el esquema no se puede corregir automáticamente, la consola mostrará una lista de errores para que los corrija manualmente.

> **Important**
>
> El editor de código de la consola de permisos verificados de Amazon siempre muestra los errores y advertencias de Cedar 4, incluso si su almacén de políticas usa Cedar 2. Puede seguir realizando actualizaciones de esquemas que no sean compatibles con Cedar 4 mediante el botón Guardar cambios o la API de permisos verificados.

Para corregir un esquema mediante la consola

1. Inicia sesión en la consola de permisos verificados de Amazon Consola de administración de AWS y ábrela en verifiedpermissions.
2. En el panel de navegación, selecciona Almacenes de políticas y, a continuación, elige el almacén de políticas que deseas consultar.
3. Elija Esquema en el panel de navegación.
4. Si tu esquema se puede corregir automáticamente, verás un banner que dice «Haz clic en 'Reparar' para previsualizar una versión compatible». Selecciona Reparar.
5. Revisa los cambios realizados en el esquema y haz clic en Vista previa del esquema actualizado.
6. Revisa el esquema actualizado y haz clic en Guardar cambios.

Si el esquema no se puede corregir automáticamente, puedes ver una lista de errores que puedes corregir tú mismo en la consola.

1. Abre la página de edición del esquema tal y como se ha descrito anteriormente.
2. Seleccione el modo JSON.
3. Coloca el puntero del ratón sobre el icono de error rojo situado en el margen izquierdo del editor de código. El mensaje de error aparece en la información sobre herramientas.

Estos son algunos de los errores más comunes que se pueden encontrar y cómo resolverlos:

no se pudo analizar el esquema de JSON: `*field-name*`

Con Cedar 2, puede incluir campos arbitrarios en partes de los esquemas, como las definiciones de tipos, incluso si no tienen ningún significado como parte de un esquema de Cedar. En Cedar 4, esto ya no está permitido. Para resolver este error, elimina el campo llamado *field-name* de tu esquema JSON. Para obtener una lista de los campos de esquema válidos, consulte la documentación de Cedar.

tipo de extensión desconocido `*extension-name*`

En Cedar 2, al declarar un atributo cuyo `type` es`Extension`, puede especificar cualquier valor para el `name` campo, independientemente de si el valor es un nombre de tipo de extensión válido o no. Ahora se trata de un error de Cedar 4. Para resolverlo, *extension-name* sustitúyalo por un nombre de tipo de extensión válido. Encontrará una lista de nombres de tipos de extensión válidos en la documentación de Cedar.

Si aún no está seguro de cómo resolver los errores de su esquema, póngase en contacto con Soporte

¿Cómo puedo hacer que mis políticas y plantillas sean compatibles con Cedar 4?

La consola de permisos verificados muestra cualquier error en la política o plantilla que la haga incompatible con Cedar 4.

Para ver los errores de una política o plantilla en la consola

1. Inicia sesión en la consola de permisos verificados de Amazon Consola de administración de AWS y ábrela en verifiedpermissions.
2. En el panel de navegación, selecciona Almacenes de políticas y, a continuación, elige el almacén de políticas que deseas consultar.

3. Seleccione Políticas o Plantillas de políticas en el panel de navegación, según corresponda.
4. Seleccione la política o plantilla incompatible.
5. Seleccione Editar.
6. Pase el ratón por encima del icono de error rojo situado en el margen izquierdo del editor de código. El mensaje de error aparece en la información sobre herramientas.

Estos son algunos de los errores más comunes que se pueden encontrar y cómo resolverlos:

Los literales vacíos están prohibidos en las políticas

En Cedar 2, puede usar la sintaxis `mySet == []` para comprobar si un conjunto está vacío. Con Cedar 4, las políticas que utilizan esta sintaxis ya no se validan con respecto a un esquema. Sustituya `mySet == []` su póliza por`mySet.isEmpty()`.

Historial de documentos de la Guía del usuario de Amazon Verified Permissions

En la siguiente tabla se describen las versiones de la documentación de Verified Permissions.

Cambio	Descripción	Fecha
Eliminación definitiva de los alias de los almacenes de políticas	Ahora puede eliminar de forma permanente los alias del almacén de políticas para eliminarlos inmediata mente, sin tener en cuenta el período de reserva de 24 horas`PendingDeletion` .	17 de abril de 2026
Nombres de políticas y nombres de plantillas de políticas	Ahora puede asignar nombres a las políticas y plantillas de políticas para hacer referenci a a ellas mediante nombres descriptivos.	4 de marzo de 2025
Alias del almacén de políticas	Ahora puede crear alias de almacenes de políticas para hacer referencia a los almacenes de políticas mediante nombres descripti vos.	26 de febrero de 2025
Nuevas políticas AWS gestionadas	Ahora puede usar las políticas `AmazonVerifiedPerm issionsReadOnlyAcc ess` IAM administradas `AmazonVerifiedPerm issionsFullAccess` y las políticas con permisos verificados.	11 de octubre de 2024

Fuentes de identidad del OIDC	Ahora puede autorizar a los usuarios de los proveedor es de identidad de OpenID Connect (OIDC).	8 de junio de 2024
Autorización por lotes con tokens de origen de identidad	Ahora puede autorizar a los usuarios de un grupo de Amazon Cognito usuarios en una sola solicitud de `BatchIsAuthorizedW ithToken` API.	5 de abril de 2024
Crear un almacén de políticas con API Gateway	Ahora puede crear un almacén de políticas a partir de una API y un grupo Amazon Cognito de usuarios existentes.	1 de abril de 2024
Conceptos y ejemplos de contexto	Se agregó información sobre el contexto en las solicitudes de autorización con permisos verificados.	1 de febrero de 2024
Conceptos y ejemplos de autorización	Se agregó información sobre las solicitudes de autorización con permisos verificados.	1 de febrero de 2024
AWS CloudFormation integrati on	Verified Permissions permite crear fuentes de identidad , políticas, almacenes de políticas y plantillas de políticas en CloudFormation.	30 de junio de 2023
Versión inicial	Versión inicial de la Guía del usuario de Amazon Verified Permissions	13 de junio de 2023

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.