GSA | Notes from the Field

Google Search Appliance

Seguridad

Mayo de 2014

Seguridad

La seguridad es clave a la hora de diseñar y de implantar soluciones que integran datos de distintas fuentes para la búsqueda para empresas. Puede convertirse en uno de los aspectos más complejos para este tipo de proyectos, sobre todo en lo referente a la Intranet, donde la seguridad acostumbra ser un requisito fundamental. Es importante destinar suficiente tiempo de calidad a esta área.

En este documento se ofrece información sobre las consideraciones que hay que tener en cuenta para crear un modelo de los requisitos de seguridad y transformarlos en la solución definitiva. Es fundamental que desde el principio se conozcan las necesidades del proyecto porque la seguridad es una de las áreas más complicadas de cambiar una vez que se ha empezado a implementar otras fases.

Acerca de este documento

La información y recomendaciones que se exponen en este documento se recopilaron gracias al trabajo que realizamos con distintos clientes y entornos del sector. Nos gustaría agradecer a nuestros clientes y partners que hayan compartido sus experiencias y puntos de vista con nosotros.

Temas tratados	En este documento se analizan todas las opciones de implantación de Google Search Appliance (GSA) para que pueda conocer mejor los protocolos de seguridad del producto. Dado que complementa la documentación de producto de GSA, encontrará referencias a los documentos del producto, donde se ofrecen descripciones detalladas sobre cómo configurar las funciones.
Público objetivo principal	Esta guía está dirigida a las personas que participen en un proyecto de búsqueda para empresas y se encarguen de los requisitos de seguridad, ya sea para recopilarlos, para diseñar el proyecto o para implantar la solución final. La información que contiene este documento puede resultar útil para las personas que ocupen las siguientes funciones: ● Director de proyecto ● Director técnico de proyecto ● Desarrollador ● Administrador de GSA
Entorno de TI	GSA configurado con distintos mecanismos de autenticación y de autorización para la búsqueda segura.
Fases de implementación	Diseño de la configuración de seguridad para GSA.
Otros recursos	● La documentación de producto de GSA proporciona información completa sobre GSA. ● Infórmese sobre cómo configurar la seguridad de GSA en el Centro de asistencia. ● Learngsa.com proporciona recursos formativos para GSA. ● El Portal de asistencia de Google for Work proporciona acceso a la asistencia de Google. ● Notas sobre GSA: introducción a la integración de contenido ● Notas sobre GSA: manual sobre los casos de implementación

Índice

Capítulo 1 Cómo diseñar la seguridad en GSA

Descripción general

Los proyectos de búsqueda para empresas incluyen datos de distintas fuentes para que los usuarios puedan encontrar la información fácilmente. En la mayoría de los casos, sobre todo en proyectos de Intranet, el acceso a los documentos en las aplicaciones fuente se encuentra protegido. Para mostrar resultados relevantes y seguros a los usuarios, el motor de búsqueda empresarial debe aplicar las mismas políticas de autorización que las fuentes en las que se almacenan los documentos.

El dispositivo de búsqueda hace las veces de concentrador: es donde se indexa el contenido procedente de las distintas fuentes para que el usuario pueda acceder a la información que necesita. El dispositivo debe confiar en los mismos protocolos de seguridad que las aplicaciones. Si en el proyecto de búsqueda para empresas se incluye la indexación de contenido protegido, deberá invertir tiempo durante la fase de diseño para crear un modelo de las relaciones de seguridad entre las fuentes de contenido y Google Search Appliance.

Antes de empezar la implantación de la seguridad en GSA, dedique un tiempo a analizar toda la integración y la arquitectura de referencia. Dado que probablemente su empresa ya cuenta con políticas y protocolos de seguridad internos, deberá estudiar cuáles son las mejores opciones de seguridad en un entorno de búsqueda. Deberá, además, diseñar un modelo de seguridad para el dispositivo de búsqueda que sea
igual en todas las fases del proyecto.

En este capítulo se explican los procesos clave de la búsqueda segura de GSA y cómo debería plantearse el diseño en general.

La búsqueda segura se divide en tres procesos distintos, aunque relacionados:

Adquisición de contenido protegido	Es el mecanismo que GSA utiliza para adquirir la fuente de contenido protegido. GSA tiene que superar la protección de la fuente de contenido para poder acceder. Es parte integral de la adquisición de contenido, pero debe considerarse parte del diseño de seguridad.
Autenticación al presentar los resultados	Se trata del mecanismo que usa GSA para identificar a los usuarios finales. Puede ser uno o varios de los protocolos de autenticación de Internet. Es la comunicación que se establece entre GSA y el cliente (navegador).
Autorización al presentar los resultados	Es el proceso que emplea GSA para comprobar si un usuario que realiza búsquedas tiene acceso a los resultados búsqueda.

Adquisición de contenido
Una vez que haya creado un modelo de la información sobre las fuentes de contenido, podrá diseñar los mecanismos de autenticación que utilizará GSA para integrar con cada una de las fuentes protegidas. En esta fase del diseño del proyecto se crea un modelo de la integración entre el dispositivo de búsqueda
y los sistemas de la empresa. El dispositivo de búsqueda permite el uso de varios mecanismos de autenticación a la vez para así poder admitir distintas aplicaciones a la hora de adquirir contenido. En el proceso se suele usar un sistema o una cuenta de superusuario con amplio acceso a la fuente de contenido para que GSA pueda indexar los documentos.

Autenticación al presentar los resultados
Se trata de la integración entre el dispositivo de búsqueda y el usuario final. El protocolo de autenticación puede coincidir con el que utilice una de las fuentes de contenido. A veces se necesitan varios protocolos de autenticación para poder autorizar distintas fuentes de contenido. Sin embargo, siempre habrá que hacerse las preguntas siguientes:

- ¿Qué protocolos de autenticación tiene el cliente en su entorno?
- ¿Cómo puedo minimizar los mecanismos de autenticación que se usan al presentar los resultados? ¿Puedo reducirlos a solo uno?
- ¿Qué puedo hacer para minimizar el impacto en los usuarios finales? La autenticación, ¿puede ser silenciosa?

Autorización al presentar los resultados
Cada fuente de contenido tiene sus propias políticas de seguridad e infraestructura para autorizar el acceso a la información que contiene. A partir de la información que haya recopilado de las fuentes de contenido deberá seleccionar los mecanismos de autorización en función de las respuestas a estas preguntas:

- ¿Qué mecanismos de autorización se pueden aplicar a la fuente de contenido en cuestión?
- ¿Cuáles son los mecanismos que ofrecen mejores resultados?
- ¿Qué debe implementarse en el proceso de adquisición de contenido para integrar este mecanismo?

Aunque la autenticación al presentar los resultados ocurre antes de la autorización durante la presentación de resultados, PRIMERO debería evaluar las opciones de autorización. Por lo general, lo que exige la autorización marca los mecanismos de autenticación que deberían tenerse en cuenta. En cualquier caso, estos tres procesos están relacionados entre sí y deberá tener en cuenta qué implica cada una de las decisiones que tome.

Recopilación de información

Google le recomienda que haga lo siguiente en la fase de análisis inicial:

- Analiza y comprende los requisitos relacionados con la seguridad, incluso necesidades que puedan surgir más adelante y que no formen parte del ámbito del proyecto, pero que puedan valer para una fase posterior.

- Si uno de los requisitos es la autenticación silenciosa, asegúrese de que es viable antes de adoptarla.
- Identifique los mecanismos de seguridad que haya certificado su empresa. ¿Tiene un sistema de inicio de sesión único? ¿Está habilitado Kerberos?

Con ayuda de la tabla siguiente, diseñe un modelo de cada una de las fuentes de contenido. En el campo de mecanismos de seguridad, incluya información sobre la seguridad.

Información del sistema	Nombre del sistema y nombre del producto subyacente
Descripción	Descripción ampliada
Tipo de contenido	Por ejemplo, ¿son documentos de herramientas ofimáticas, páginas web, registros de bases de datos...?
Tamaño del contenido	Recuento de documentos: si el contenido es reducido, es posible que pueda aplicarse el enlace en tiempo de ejecución para la autorización al presentar los resultados
Autenticación al presentar los resultados	¿El servidor de contenido utiliza Windows Integrated Authentication?
	¿El servidor de contenido está integrado en un sistema de inicio de sesión único?
	Si el servidor de contenido tiene un directorio de usuario propio en formato de base de datos o LDAP, ¿los nombres de usuario están sincronizados con el directorio de toda la empresa?
Autorización al presentar los resultados	¿Responde bien el servidor de contenido? Si no es así, el enlace en tiempo de ejecución probablemente no sea una buena opción
	¿Los permisos de los documentos son bastante amplios o, por el contrario, son muy restrictivos? Si son muy restrictivos, el enlace en tiempo de ejecución probablemente no sea una opción acertada
	¿Hay una API que permita saber si un usuario tiene acceso a ciertos documentos a partir del nombre de usuario y de los ID de documento? Si es así, se podrá aplicar la autorización CONNECTOR o SAML
	¿Hay alguna forma de saber qué grupos, funciones y usuarios tienen acceso a cada uno de los documentos? Si es así, probablemente la autorización ACL será la más adecuada.

Adquisición de contenido

La adquisición suele presentar los formatos siguientes. Tenga en cuenta que deberá usarse el protocolo de autenticación compatible con la fuente de contenido. Sin embargo, la adquisición de contenido suele permitir el uso de distintos mecanismos de autorización.

	Mecanismos de autorización posibles para la presentación de resultados	Notas
Rastreo directo	Autorización ACL, SAML y de solicitud HEAD	Quizá haya que desarrollar un servidor proxy personalizado para procesamientos más extensos
Feeds	Autorización ACL, SAML y de solicitud HEAD	Una única implementación simple y personalizada
Conectores	Autorización ACL y CONNECTOR	Podría tratarse de un conector estándar o de un conector que puede personalizarse

Identidades únicas o múltiples

Tras examinar todas las fuentes de contenido, debería poder responder a una pregunta muy importante: ¿es suficiente con un grupo de credenciales (opción predeterminada)? Esto determinará qué modelo de autenticación debe utilizarse para presentar resultados. Si hay varias identidades por usuario, probablemente habrá que definir varios grupos de credenciales. Si hay protocolos de autenticación distintos para las diversas fuentes de contenido, eso no significa que haya varias identidades. Por ejemplo, una fuente de contenido podría usar una autenticación basada en formularios y otra podría usar Kerberos. Sin embargo, si se usa el mismo Active Directory como directorio de usuario en ambos sistemas, habrá una sola identidad por usuario. GSA puede necesitar varias identidades solo si la información de usuario se almacena en repositorios diferentes. Sin embargo, hay dos excepciones:

- Si un directorio de usuario es la réplica de otro, sigue habiendo una identidad por usuario. Por lo tanto, bastará con un grupo de credenciales. Por ejemplo, si Documentum está integrado con Active Directory, una posibilidad es replicar todos los usuarios en la base de datos de Documentum.
- Si los nombres de usuario de dos directorios de usuario coinciden exactamente, bastará con un grupo de credenciales siempre y cuando inserte un servicio de traducción de identidad de usuario, quizá en el proceso de autorización.

Selección de un mecanismo de autorización

La autenticación y la autorización de presentación de resultados son dos procesos estrechamente relacionados. Tal y como se ha mencionado anteriormente, aunque la autenticación al presentar los resultados ocurre antes de la autorización durante la presentación de resultados, PRIMERO debería evaluar las opciones de autorización. Es una cuestión muy importante que vale la pena reiterar. En este capítulo se describe en profundidad la conexión entre estos dos procesos.

La autorización se aplica siempre según la fuente de contenido. El objetivo de la autorización es que los usuarios vean en los resultados de búsqueda lo que tengan permiso para ver. Aparte de este objetivo final, el criterio más importante a la hora de seleccionar el mecanismo de autorización es el **rendimiento**. Implica lo siguiente:

- Los resultados de búsqueda deben mostrarse lo más rápidamente posible para así ofrecer a los usuarios finales el mejor servicio posible. Según estudios sobre la facilidad de uso, si la búsqueda es demasiado lenta, el usuario simplemente deja de hacer búsquedas y el uso de esta función disminuye.
- El rendimiento debe ser bueno para que el tiempo de espera no se agote y puedan mostrarse resultados relevantes al usuario. Si para ciertos resultados el tiempo de espera de la decisión de autorización se agota, los resultados tendrán una decisión de autorización indeterminada y, por lo tanto, no se mostrarán en la lista de resultados de búsqueda.
- Si se utiliza la autorización con enlace en tiempo de ejecución, deberá minimizar el efecto en el rendimiento del servidor de contenido.

En los proyectos de desarrollo, si ya hay un conector que haya suministrado Google o uno de sus partners, la autorización ya viene marcada por el diseño de dicho conector. Deberá seleccionar un mecanismo de autorización únicamente en estos casos:

- Varios proveedores ofrecen conectores que usan mecanismos de autorización diferentes. Muchos serán los factores que determinen qué conector habrá que usar, como los costes, y el mecanismo de autorización es solo uno de ellos.
- A veces, un conector admite varios mecanismos de autorización. Por ejemplo, el conector de Google Search Appliance para SharePoint admite tres mecanismos: ACL por URL, CONNECTOR y de solicitudes HEAD.
- En caso de que no haya conector, deberá desarrollar un código personalizado para poder integrar el contenido protegido. En este caso, deberá tener en cuenta todas las opciones.

A continuación analizaremos la autorización según el orden de preferencia para el rendimiento. GSA procesa la autorización a partir de dos enfoques básicos:

- Autorización con enlace en tiempo de compilación
- Autorización con enlace en tiempo de ejecución

Por lo general, el enlace en tiempo de compilación agiliza el proceso de autorización en GSA en comparación con el enlace en tiempo de compilación, pero eso no significa que deba usarse siempre en todas las fuentes de contenido.

Autorización con enlace en tiempo de compilación

Si se usa el enlace en tiempo de compilación, el dispositivo de búsqueda se encarga de gestionar la autorización. En el caso del enlace en tiempo de compilación, hay que indicar a GSA las reglas de autorización. No es necesario que se comunique con un componente de seguridad externo, como la fuente de contenido, al publicar los resultados para comprobar si el usuario está autorizado a acceder a un determinado documento.

GSA admite los siguientes dos tipos de ACL:

ACLs por URL

Con este tipo de ACLs, cada uno de los documentos del índice puede tener sus propias reglas de autorización. Se puede añadir una ACL por URL a un documento a través de feeds, de metadatos en el cuerpo de un documento HTML o de cabeceras HTTP personalizadas. Las ACL por URL pueden incluir tanto usuarios como grupos. En general, es preferible usar ACLs por URL porque admiten mejor las ampliaciones de número de documentos y porque ofrecen mejores resultados.

Aspectos que hay que tener en cuenta cuando se usen ACLs por URL:

- Este tipo de ACLs es muy útil si tiene reglas de autorización muy específicas y quiere que las respuestas de autorización sean rápidas. Con ACLs, la rapidez en las autorizaciones es fundamental para funciones de GSA como la navegación dinámica, el filtrado de directorios duplicados y los clústeres de resultados dinámicos.

- Las ACL por URL presentan alguna dificultad a la hora de resolver miembros de grupo en el dispositivo de búsqueda. En algunos casos, GSA puede gestionar esta resolución, por ejemplo, saber si esos grupos se encuentran en un directorio LDAP como Active Directory. También puede crear procesos personalizados para transferir grupos al dispositivo de búsqueda. En la versión 7.2 se incorpora una base de datos de grupos como función beta que ofrece una integración incluso mayor.
    - El conector Active Directory Groups de Google Search Appliance se suministra para resolver grupos de dominios únicos o múltiples de Active Directory.

- También se produce un retraso entre que se cambia una configuración de seguridad en la plataforma fuente y se notifica de ello al dispositivo de búsqueda.

- Es posible configurar el número máximo de objetos principales que se puede adjuntar a un documento. El valor predeterminado es 10.000 y el máximo, 100.000.
    - A continuación se expone un caso con condiciones desfavorables en el que el filtrado de ACL ha tenido buenos resultados (subsegundo):
        - Filtrado de 10.000 URLs
        - Cada URL incluye 10.000 elementos en la ACL
        - El usuario de búsqueda pertenece a 1.000 grupos pero no tiene acceso a ningún documento, por lo que GSA debe filtrar exhaustivamente todas las URL que coincidan con el término de búsqueda.

Políticas ACL
El objetivo de la política ACL es proteger los patrones de URL en lugar de cada URL por separado. Por este motivo puede agrupar muchos documentos. Puede configurar políticas de ACL según patrones de URL a través de la Consola del administrador de GSA y del API de políticas ACL. Utilice las políticas de ACL cuando el número de reglas de autorización sea bajo y una sola regla pueda agrupar varias URL.

Aunque no se usa tanto como las ACL por URL, es una herramienta muy flexible que puede venir bien en determinados momentos. Por ejemplo, si hay que denegar a un grupo que está definido de forma global el acceso a una fuente de contenido fácilmente identificable, bastaría con definir una sola entrada de política ACL. Otro caso sería cuando el sistema de contenido utiliza reglas de permiso amplias. Por ejemplo, CA SiteMinder permite definir el control de acceso según patrones de URL. Esas reglas luego pueden traducirse fácilmente en políticas ACL.

Desde la versión 7.0 de GSA, para las políticas ACL hay que especificar el dominio, el espacio de nombres y la distinción entre mayúsculas y minúsculas.

Autorización con enlace en tiempo de ejecución

Con este tipo de autorización, el dispositivo de búsqueda no tiene información de autorización sobre el contenido protegido (es decir, ACLs). Antes de que GSA muestre los resultados de búsqueda al usuario, debe comprobar la seguridad. Para ello, se comunica con un componente de terceros y verifica si el usuario tiene permiso para leer cada uno de los documentos protegidos que se incluyen en los resultados. Dicho componente responde enviando la decisión de autorización correspondiente al dispositivo de búsqueda. Este componente de terceros podría ser la propia fuente de contenido o un servidor de autorización que se encarga de centralizar la decisión.

GSA admite los siguientes tipos de enlace en tiempo de ejecución:

Conectores
Google proporciona algunos conectores, como SharePoint o Documentum, que puede usar en sus proyectos para integrar el dispositivo de búsqueda con fuentes de terceros y seguir siendo totalmente compatibles con Google. Se ejecutan en una estructura para conectores creada por Google que se puede usar para crear conectores propios. La principal ventaja de esta plataforma para crear conectores propios es una integración máxima entre la configuración, la indexación y la seguridad con el dispositivo de búsqueda.

La estructura para conectores proporciona la interfaz SPI para que cualquier conector pueda implementar
la autorización. La interfaz funciona en modo por lotes (varios documentos en una invocación) para que las respuestas sean lo más directas posible. Existen otros conectores proporcionados por partners de Google que se basan en la estructura y emplean este procedimiento.

Autorización SAML
SAML es una estructura basada en XML que sirve para comunicar la autorización, los permisos y la información de atributo de los usuarios. Es un estándar que se puede usar para la autenticación, aunque, de manera opcional, también sirve para procesos de autorización. La SPI de autorización explica cómo puede usarse SAML para el proceso de autorización. Si se usa SAML para la autorización no significa que debamos usarlo también para la autenticación, y al revés. Ambos procesos son totalmente independientes. En este caso, el dispositivo de búsqueda envía solicitudes de autorización SAML en formato XML al servicio externo que haya configurado y el servidor responde enviando los permisos de autorización para cada documento.

Los productos de autenticación SAML estándar (IdPs) son bastante habituales, pero los proveedores de servicio de autorización no lo son tanto. SAML Bridge ofrece esta función para usar la suplantación de identidad de Kerberos para autorizar el uso de solicitudes HEAD. Se trata de una característica heredada de cuando las autorizaciones CONNECTOR y ACL no estaban disponibles. Significa que este enfoque probablemente será un proyecto personalizado que desarrolle usted mismo, en lugar de usar un producto que ya existe.

Las autorizaciones SAML se pueden gestionar por lotes de manera que el dispositivo de búsqueda envía una lista de URLs para la autorización por solicitud, que puede acelerar el proceso. Puede activar esta opción en la Consola del administrador de GSA, pero el proveedor de autorizaciones SAML debe admitirla.

Solicitudes HEAD
Por último, también es posible enviar una solicitud HEAD HTTP a la fuente de contenido para validar las autorizaciones. GSA puede enviar una solicitud HTTP usando la URL del documento y leer la respuesta HTTP de la fuente para determinar la autorización a partir de los códigos de error HTTP:

- 200: Este código básicamente significa que el usuario puede acceder al documento, por lo que el dispositivo de búsqueda lo consideraría un permiso. También se pueden definir algunas reglas de exclusión en el dispositivo de búsqueda, ya que algunas fuentes de contenido incluyen códigos de error HTTP 200, incluido un mensaje de acceso no permitido, como ocurre en algunas soluciones de portales web.

- Los demás códigos de error indican que el usuario no puede acceder al documento en cuestión.

Para comprobar el permiso de todos los resultados, se envía una solicitud HEAD por documento de forma secuencial hasta que haya un número suficiente de documentos con permiso para llenar al menos una página de resultados de búsqueda. Por eso, la solicitud HEAD es el mecanismo de autorización que peor funciona. Se suele utilizar cuando no hay forma de extraer la ACL o cuando no se pueden comprobar los permisos mediante un API.

Conectores 4.0(beta)
La estructura para conectores versión 4.0 es una herramienta nueva que se basa en una arquitectura completamente distinta de las versiones anteriores. Tenga en cuenta que todavía está en fase beta. Las funciones de seguridad que ofrece también son distintas de las otras versiones. Estas son algunas de las diferencias más destacadas en cuanto a la seguridad:

- Se puede integrar un conector como mecanismo de autenticación y de autorización. El protocolo de comunicación entre el dispositivo y el conector ya no es XML de propiedad. En su lugar, se usa SAML como mecanismo de intercambio de mensajes subyacente. Un ejemplo de este conector sería el adaptador de autenticación de Google, que proporciona autenticación para los ID de Google. Se configura de la misma forma que un proveedor de SAML.

- Un conector integrado en la estructura 4.0 admite ACL por URL.

- Se pueden crear conectores para ofrecer la resolución de grupos a través de la autenticación SAML. Sin embargo, en GSA 7.2, la mejor opción es la resolución de grupos integrada (beta). La resolución de grupos a través de SAML forma parte de la autenticación SAML, a diferencia de la estructura para conectores anterior donde los conectores únicamente pueden resolver grupos si la autenticación la realiza otro mecanismo.

Cómo seleccionar el mecanismo de autenticación

En una implementación suele haber varios mecanismos de autenticación disponibles. Tal y como se ha mencionado en el primer capítulo, el objetivo principal es usar el menor número posible de mecanismos de autenticación. A menudo suele haber otro requisito, la autenticación silenciosa. No todos los mecanismos de autenticación funcionan con todos los mecanismos de autorización. Los mecanismos de autorización
se dividen en dos categorías:

- **El ID del usuario es necesario**
- **El ID del usuario no es necesario**

Todos los mecanismos de autorización necesitan el ID del usuario salvo las solicitudes HEAD. En la tabla siguiente se incluyen los mecanismos de autenticación que necesitan el ID de usuario:

	Mecanismo de autenticación que requiere el ID de usuario
Básica HTTP/NTLM	Aparece como básica HTTP/NTLM. Sin embargo, estos son los protocolos de autenticación que se utilizan para comprobar las credenciales de usuario que se transmiten entre GSA y un servidor. Para el usuario final es una autenticación basada en formularios. Una vez que la URL de muestra configurada verifica las credenciales de usuario, el ID que introduce el usuario se considera el ID verificado.
Certificado de cliente	El nombre completo del certificado se transfiere como ID verificado.
Kerberos	El ID de usuario de Windows que se extrae de la incidencia de Kerberos se usa como el ID de usuario verificado.
Autenticación SAML	El IdP SAML envía el ID verificado en el campo "Subject".
Autenticación LDAP	El ID de usuario que ha comprobado el servidor LDAP se usa como el ID verificado.

Autenticación basada en formularios con acceso a cookies	El ID de usuario se vuelve a enviar a GSA a través de la herramienta para acceder a cookies. Para enviarlo, se necesita algo de codificación para implementar una simple página web dinámica.
Conectores	La estructura para conectores proporciona la SPI de autenticación que devuelve un ID de usuario de confianza. Sin embargo, debe implementarlo el conector, lo cual es opcional. No todos los conectores disponibles proporcionan autenticación. Las herramientas que implementan conectores incluso pueden solicitar una contraseña. Por ejemplo, File System Connector 2.x solicita el nombre de usuario y la contraseña para poder aplicar la autorización con enlace en tiempo de ejecución cuando hay reglas de rechazo en documentos.

Los mecanismos de autenticación anteriores se pueden combinar con autorizaciones ACL, CONNECTOR
y SAML. Elija el que se adapte mejor a las necesidades del cliente y sea más fácil de implantar. Tenga en cuenta también los posibles requisitos de autenticación silenciosa.

En el caso de las autorizaciones de solicitud HEAD, no podrá elegir cualquier mecanismo de autenticación porque este tipo de solicitudes se envían desde GSA a la fuente de contenido y no desde el navegador cliente a GSA. Según el protocolo de autenticación que utilice la fuente de contenido, GSA deberá obtener distintas credenciales durante el proceso de autenticación de usuario.

	Mecanismo de autenticación que no requiere el ID de usuario (solicitudes HEAD)
Cookie	Este es el caso más habitual: el dispositivo de búsqueda reenvía las cookies de usuario para validar los derechos de acceso. Es necesaria la autenticación basada en formularios. Puesto que no se exige el ID de usuario, tampoco es necesario el acceso a las cookies. La regla se encuentra configurada en **Mecanismos de autenticación de acceso universal > Cookies.**
HTTP	Debe configurarse una regla básica HTTP/NTLM. De hecho, la comunicación entre el navegador del usuario final y el dispositivo de búsqueda es a través de la autenticación basada en formularios, en lugar del protocolo de autenticación básica HTTP. La regla se encuentra configurada en **Mecanismos de autenticación de acceso universal > HTTP**.
Kerberos	Debe configurarse una regla básica HTTP/NTLM. La comunicación entre el navegador del usuario final y el dispositivo de búsqueda es a través de la autenticación basada en formularios. La regla se encuentra configurada en **Mecanismos de autenticación de acceso universal > Kerberos**.

Asignación de la autenticación a la autorización

En la mayoría de los casos, deberá elegir un mecanismo de autenticación para verificar los usuarios. Se pueden configurar distintos mecanismos de autenticación pero, ¿qué implica cada uno de ellos? ¿para qué sirven? Recuerde estas reglas:

- Si hay varios grupos de credenciales, deberá seleccionar un mecanismo para cada uno. Pueden ser del mismo tipo, por ejemplo, dos autenticaciones basadas en formularios o dos autenticaciones SAML. O pueden ser mecanismos distintos: uno mediante formularios y, el otro, Kerberos.

- Si lo prefiere, puede configurar varios mecanismos de autenticación para el mismo grupo de credenciales, aunque no es tan habitual. Sin embargo, si se usa un conector para la resolución de grupos, un mecanismo de autenticación (probablemente, silencioso) verifica el usuario que realiza la búsqueda y el conector resuelve la pertenencia al grupo para la autorización ACL. Analizaremos este aspecto con mayor profundidad en capítulos posteriores.

- Se activarán todos los mecanismos de autenticación. Si se definen dos mecanismos de autenticación para un mismo grupo de credenciales, el segundo se activará aunque el primero haya verificado correctamente el ID de usuario.

Para asignar la autenticación a la autorización, el dispositivo de búsqueda utiliza una función denominada "autorización flexible", parecida a una tabla de enrutamiento para los mecanismos de autorización. Permite al administrador configurar el proceso de autorización para documentos en función de los patrones de URL, según le convenga para la implementación que esté llevando a cabo. La autorización flexible se gestiona a través de la configuración de reglas de autorización. Esta sería una posible regla: el contenido al que debe aplicarse la regla (definido por el patrón de URL), una identidad que asigne la regla a una regla de credenciales o a un mecanismo de autenticación y otra información específica del mecanismo de autorización.

En la mayoría de los casos no es necesario cambiar la configuración de la autorización flexible; basta con usar los ajustes predeterminados. Es una tabla de enrutamiento donde pueden combinarse los mecanismos de autenticación y de autorización, aunque hay ciertas reglas que no pueden usarse juntas:

- ACL por URL
    - Las ACL forman parte del índice y no se pueden añadir o quitar de forma inmediata. Si las URL no incluyen ACLs, no se puede usar el mecanismo ACL por URL. El grupo de credenciales asociado a las ACL también se determina durante el tiempo del índice, pero no se puede cambiar en la configuración de la autorización flexible.
    - Si la regla ACL por URL se define como la primera regla, la autorización se produce en el índice una vez identificados los resultados coincidentes. Por eso este tipo de mecanismo es más eficaz que otras autorizaciones. Y también por eso aparece antes que la autorización mediante caché de forma predeterminada.

    - Si define un patrón de URL específico en lugar de "/" o desplaza la regla por debajo de otras reglas de autorización, el administrador de seguridad se encargará de realizar la autorización. En este caso, el resultado será más ineficaz porque las ACL por URL se evalúan fuera del índice.

- CONNECTOR
    - Si el contenido debe autorizarse mediante la autorización CONNECTOR, las URL deben empezar por googleconnector://.

Una vez que GSA haya autenticado a un usuario a través de un mecanismo de autenticación configurado, se aplicará la autorización a los documentos por orden de definición de la tabla de autorización flexible según el patrón de URL del documento. Si hay más de un mecanismo de autorización que afecte al documento, GSA pasará por todas las reglas correspondientes por orden hasta que se devuelva el estado de permiso o de rechazo, PERMIT o DENY. Por ejemplo, si un conector envía documentos con ACLs,
la regla ACL por URL se evaluará en primer lugar. Si se devuelve PERMIT o DENY, este será el resultado definitivo. Sin embargo, si se devuelve un resultado indeterminado o INDETERMINATE, se usará la regla CONNECTOR para evaluar los documentos.

Resumen

En este capítulo se ha explicado el proceso para diseñar la seguridad de un proyecto de búsqueda empresarial con Google Search Appliance. Para ello, hay que conocer bien el área de seguridad de la empresa, así como las fuentes de contenido relacionadas que formarán parte del proyecto. A continuación, se incluye un resumen del proceso para diseñar la solución:

- Dedique tiempo antes de empezar a analizar las fuentes de contenido: ¿cómo se adquirirán las fuentes de contenido? ¿qué mecanismo de autenticación se utiliza?, etc.
- Averigüe cuántos grupos de credenciales se necesitarán.
- Decida qué mecanismo de autorización es el más adecuado para cada una de las fuentes de contenido.
- Determine cuántos tipos de mecanismos de autenticación va a necesitar como mínimo.
    - Siempre que sea posible, aplique la autenticación silenciosa.
    - Siempre que sea posible, utilice componentes estándar.
    - Estos mecanismos de autenticación, ¿admitirán la autorización de la fuente de contenido correspondiente?
- Configure la opción "Mecanismos de autenticación de acceso universal".
- Cambie las reglas de la autorización flexible cuando lo considere oportuno.

Capítulo 2 Uso de funciones estándar

En este capítulo profundizaremos en algunos de los métodos de autenticación y de autorización. Veremos también los entornos más habituales que admite Google Search Appliance y los productos relacionados que ofrece Google. Nos centraremos en casos en los que no es necesario escribir código.

Autenticación silenciosa

El objetivo de la seguridad de las TI es proteger las aplicaciones y los datos, y ofrecer información precisa a los usuarios de forma segura. Pero también es importante que los mecanismos de control de acceso tengan unas consecuencias mínimas para el usuario. Por ejemplo, si un componente de confianza ya ha autenticado a un usuario, las aplicaciones deberían confiar en ese proceso y evitar así volver a solicitar las credenciales al usuario o verificar su identidad. Este es el concepto que entraña la autenticación silenciosa: verificar la identidad del usuario en GSA sin tener que solicitarle datos ni pedirle que inicie sesión otra vez.

La autenticación silenciosa se puede implementar para un servicio de búsqueda, igual que para cualquier otra aplicación de una empresa. Hay varios mecanismos de autenticación que permiten proporcionar un servicio de autenticación silenciosa; protocolos como Kerberos o NTLM, o bien aplicaciones empresariales tales como el sistema de inicio de sesión único.

Antes de implementar la autenticación silenciosa para el entorno de búsqueda, responda a las preguntas siguientes:

- ¿De qué opciones de autenticación silenciosa dispone en su empresa? ¿Hay alguna que tenga preferencia?
- Si existe más de una opción para la autenticación silenciosa, (p. ej., autenticación basada en formularios y Kerberos), ¿se encargan de gestionar las distintas identidades y credenciales de usuario que se necesitan para la autorización? Debe decidir si basta con usar una de estas opciones o si debe usar las dos. Piense también en si una puede confirmar la identidad de la otra.
- ¿Hay varios dominios de autenticación? Por ejemplo, distintos dominios de Windows para Kerberos. Esta información también es importante para crear un modelo del proceso de autorización.
- ¿Qué aplicaciones o fuentes de contenido que debe integrar en el motor de búsqueda utilizan también el mecanismo de autenticación silenciosa? Quizá pueda aprovecharlas.

El dispositivo de búsqueda puede integrarse sin necesidad de configurar ajustes con los protocolos o sistemas de autenticación siguientes:

Autenticación basada en formularios o en cookies

La autenticación basada en formularios o en cookies es el proceso que activa una cookie de sesión, generalmente de un sistema de inicio de sesión único. Este proceso podría ser silencioso si ya se ha autenticado al usuario antes de acceder al dispositivo de búsqueda. De lo contrario, se le pediría al usuario credenciales para crear las cookies de sesión adecuadas que proporcionan el servicio de inicio de sesión único. En la sección sobre casos de autenticación basada en cookies de la documentación de GSA encontrará información técnica detallada sobre la integración en un sistema de inicio de sesión único. Si además es necesario enviar un ID de usuario al dispositivo de búsqueda, deberá implementar un proceso para acceder a las cookies.

Kerberos

El protocolo Kerberos es el que se usa de forma predeterminada en las redes Windows. Se puede configurar el dispositivo de búsqueda para que se habilite Kerberos y el proceso de autenticación sea transparente para los usuarios.

SAML

Muchos sistemas de inicio de sesión único admiten el protocolo SAML y ofrecen un proceso de autenticación silenciosa. El protocolo SAML es la manera que tiene un servicio externo de confirmar con seguridad la identidad del usuario a GSA. El mecanismo de autenticación real entre el usuario y este servicio seguirá pasando por los protocolos de autenticación estándar, como Kerberos, NTLM o basados en cookies. Es muy poco habitual que tenga que escribir un proveedor de identidades SAML (IdP SAML) desde cero. Es mucho más habitual integrar GSA con un IdP SAML que ya esté implementado en la red del cliente.

Certificados cliente

Esta no es una situación habitual. Sin embargo, si el entorno del usuario cuenta con certificados cliente, también se puede configurar el dispositivo de búsqueda para que autentique a los usuarios mediante certificados X.509 que, a su vez, también ofrecen autenticación silenciosa a los usuarios.

SAML

El dispositivo de búsqueda admite la integración con SAML, un estándar de seguridad que permite crear procesos de autenticación específicos y separados del motor de búsqueda. Si crea un proveedor de autenticación SAML, podrá codificar la lógica de autenticación que necesite. Si este proceso externo autentica correctamente al usuario, la identidad del usuario se vuelve a enviar al dispositivo de búsqueda.

Puesto que SAML es un estándar de seguridad, se admite en algunas soluciones de autenticación comercializadas y de código abierto, y algunos sistemas de inicio de sesión único ofrecen una interfaz SAML. Compruebe si las soluciones de autenticación de su empresa ya incluyen una interfaz de autenticación de este tipo para simplificar la integración con el dispositivo de búsqueda. Si es así, no tendrá que crear dicho servicio.

Tenga en cuenta que también es posible configurar un proceso de autorización SAML tal y como se describe en el Capítulo 3, pero es independiente de si la autenticación SAML está configurada o no.

Consulte la documentación de producto de GSA para saber cómo configurar SAML en el dispositivo de búsqueda.

Enlace en tiempo de compilación con ACL por URL

Cuando utilice ACLs para el proceso de autorización, tenga en cuenta que todos los componentes que forman una ACL deben coincidir con la identidad resuelta para que la verificación de ACL sea válida: dominio, objeto principal de usuario, objetos principales de grupos, espacios de nombres para los objetos principales de grupo y usuario, distinción entre mayúsculas y minúsculas especificada y el tipo de ACL (PERMIT/DENY).

Resolución de grupos

A diferencia de otros mecanismos de autorización, la autorización ACL incluye un paso más: resolución de grupos para un ID de usuario verificado. El concepto de resolución de grupos es muy importante en tanto que GSA admite el enlace en tiempo de compilación con ACL. Puesto que un usuario puede ser miembro de varios grupos en un sistema de gestión de identidades, debe proporcionarse el mismo modelado de necesidades de identidad en GSA. Tras la autenticación, GSA almacena ID de usuario junto con los grupos a los que pertenece el usuario. Los grupos se pueden resolver mediante estas cinco opciones:

- **Base de datos de grupos**(beta). A partir de la versión 7.2, el dispositivo de búsqueda incluye una base de datos interna que almacena ACLs. Es una función que todavía está en fase beta, con funciones y una escalabilidad limitadas. Es necesario insertar en el dispositivo los usuarios que forman parte de grupos, igual que se insertan en el índice del dispositivo los documentos.
- **Conectores**. La estructura para conectores proporciona una interfaz para resolver grupos. El desarrollador del conector es quien decide si implementar ACL por URL o la resolución de grupos. De entre todos los conectores que admite Google, SharePoint, Active Directory Groups y Documentum son los que ofrecen esta función.
- **LDAP**. La autenticación LDAP puede resolver grupos LDAP anidados. No se recomienda para Active Directory (en su lugar, utilice el conector Active Directory Groups), pero se puede usar para otros servidores LDAP.

Las tres opciones anteriores pueden usarse ÚNICAMENTE para resolver grupos cuando la autenticación la realiza otro mecanismo. Las dos opciones siguientes resolverán grupos durante el proceso de autenticación; no se pueden usar solo para resolver grupos.

- **Acceso a cookies**. Los grupos se pueden devolver en una cabecera personalizada junto con el ID de usuario. Debe incluirse en el proceso de autenticación de cookies.
- **SAML**. Se pueden devolver grupos durante el proceso de autenticación SAML; debe incluirse en el proceso de autenticación SAML.

Estos dos mecanismos se suelen usar en implantaciones que exigen un desarrollo personalizado. En el capítulo siguiente encontrará documentación más específica sobre esta cuestión.

Nombre completo

GSA admite nombres completos de ACL. El concepto de espacio de nombres se introdujo para evitar conflictos de nombre entre usuarios y grupos de las distintas fuentes del índice. Veamos un ejemplo:

El usuario John Smith tiene dos identidades y hemos configurado dos grupos de credenciales, jsmith en CG1 y johns en CG2. En el índice las ACL que pertencen a John Smith podrían asociarse a cualquiera de esas dos identidades. Tiene que haber una forma de diferenciarlas. Por eso se introdujo el concepto de espacio de nombres.

Si el ámbito principal es usuario, el espacio de nombres equivale al grupo de credenciales. En las ACL, el objeto principal debe ser una de las siguientes opciones:

jsmith en el espacio de nombres CG1
o
johns en el espacio de nombres CG2

Sin embargo, si el ámbito principal es grupo, el espacio de nombres no tiene que ser el mismo que el grupo de credenciales del usuario. La comprobación de permiso será válida siempre y cuando el espacio de nombres de los grupos resueltos coincida con lo que se ha definido en la ACL del índice. Veamos un ejemplo:

La primera identidad de John Smith, jsmith, es de Active Directory de toda la empresa. Por supuesto, hay grupos de Active Directory a los que jsmith pertenece. Supongamos que una de las fuentes de contenido es Plone, que está integrada en Active Directory, pero tiene definidos sus propios grupos. ¿Cómo puede evitarse el conflicto de nombres cuando hay grupos con los mismos nombres tanto en Active Directory como en Plone? Los grupos de Active Directory tendrán el espacio de nombres CG1. Podemos asignar otro espacio de nombres a los grupos de Plone, como plone_space. Las ACL del índice incluirán las entradas siguientes:

```
<principal namespace="CG1" scope="user" access="permit">jsmith</principal>
…
<principal namespace="CG1" scope="group" access="permit">authors</principal>
…
<principal namespace="plone_space" scope="group"
access="deny">authors</principal>
```

Se aplicarán los permisos correspondientes siempre que puedan resolverse los grupos correctos para jsmith durante la resolución de grupos después de la autenticación:

CG1:jsmith pertenece a los grupos:
CG1:authors, plone_space:authors

Análisis de dominios

Los nombres de dominio se usan con frecuencia en las credenciales y en los grupos de usuarios. El dispositivo de búsqueda dispone de un campo aparte para los *dominios* cuando el objeto principal se almacena en los casos siguientes:

- Una vez autenticado el usuario, el ID verificado y resuelto y los grupos asociados incluyen el nombre de usuario y el nombre del dominio.
- El objeto principal de las ACL de los documentos para usuarios y grupos incluye el nombre de dominio y el nombre del objeto principal.

A partir de los distintos protocolos de autenticación, los usuarios verificados pueden adoptar diferentes formatos:

- david@**google**.com
- **google**\david

El dispositivo de búsqueda analiza estos formatos sistemáticamente y extrae el nombre de dominio y el nombre de usuario durante el proceso de autenticación y la indexación de ACL. En los dos ejemplos anteriores, el dispositivo extraería el dominio **google**.

Enlace en tiempo de ejecución para ACLs

Si utiliza ACLs para gestionar el acceso a documentos de GSA, quizá le interese configurar una alternativa al enlace en tiempo de ejecución en caso de que las ACL del índice no estén totalmente sincronizadas con la fuente de contenido por cuestiones relacionadas con el tiempo. Cuando la función alternativa al enlace en tiempo de ejecución de la autorización flexible se habilite, GSA solo aceptará la respuesta DENY para los mecanismos POLICY y ACL por URL. En el caso de PERMIT y de INDETERMINATE, GSA aplica las reglas subsiguientes hasta que una de ellas devuelva una decisión distinta a INDETERMINATE. Si esto
no ocurre, el resultado no se mostrará al usuario.

Conectores que usan ACL por URL

Espacio de nombres local

Con la estructura para conectores se introdujo el concepto de espacio de nombres local. Tenga en cuenta que se trata de un concepto relacionado con conectores. Según la definición de ACL, solo existe un atributo de espacio de nombres ("namespace"). En la configuración de conectores, hay dos campos para
el espacio de nombre. Uno es "Espacio de nombre global", que equivale al de grupos de credenciales del proceso de autenticación. Y el otro campo es "Espacio de nombre local", que será el nombre del conector (o el nombre de otro conector configurado, que se puede seleccionar en el menú desplegable).

Retomemos el ejemplo anterior sobre la fuente de contenido Plone. Si se crea un conector Plone según la estructura de conectores con el nombre de instancia "plone_connector", este es el aspecto que tendrán los objetos principales de ACL en los feeds que envíe el conector:

```
<principal namespace="CG1" scope="user" access="permit">jsmith</principal>
...
<principal namespace="CG2" scope="user" access="permit">johns</principal>
...
<principal namespace="CG1" scope="group" access="permit">authors</principal>
...
<principal namespace="CG1_plone_connector" scope="group"
access="deny">authors</principal>
...
```

El dispositivo de búsqueda concatena los campos "Espacio de nombre global" y "Espacio de nombre local" en la configuración del conector como el atributo "namespace" en la ACL que se envía a través de los feeds.

Cómo evitar el análisis de dominios

Tal y como se describe en la sección anterior, el dispositivo intenta interpretar el formato del objeto principal y extraer el dominio. Sin embargo, hay una excepción. Cuando las ACL se envían a través de feeds, si el atributo **principal_type** está definido en "unqualified" en un objeto principal, el dominio no se analizará y el nombre se tratará de forma literal, independientemente del formato. Este atributo y comportamiento se diseñan como opción alternativa para evitar conflictos con los nombres de grupos, principalmente como un pirateo para que el conector SharePoint siga siendo compatible con versiones anteriores. SharePoint permite definir grupos en los distintos niveles de la estructura jerárquica de un sitio web. Si se usara la función "Espacio de nombre local" del conector, habría un espacio de nombres por cada sitio. El conector de GSA para SharePoint incluye como prefijo en todos los grupos locales de SharePoint las URL de los sitios a los que pertenecen los grupos y define el atributo principal_type como "unqualified". El dispositivo de búsqueda almacena estos grupos a medida que los recibe para que no haya conflictos con los nombres de grupos de los distintos sitios. A continuación se muestra un ejemplo de grupos locales de SharePoint que se envían a GSA a través de feeds:

```
<principal principal-type="unqualified" namespace="Default_sp" case-
sensitivity-type="everything-case-insensitive" scope="group"
access="permit">[http://w2k8r2entsp1]Home Owners</principal>
```

Sin embargo, si se envía un grupo de Active Directory, tendrá este aspecto:

```
<principal namespace="Default" case-sensitivity-type="everything-case-
insensitive" scope="group" access="permit">mydomain\Home Owners</principal>
```

Conectores 4.0 (beta)

Uso de ACL por URL

Los conectores 4.0 indexan las ACL de forma distinta a las versiones anteriores:

- Las ACL no se envían por medio de feeds, sino que se indexan como cabeceras HTTP.
- Si las ACL son jerárquicas, no se deshará esta estructura. Se aplicará la herencia en todo momento.
- Cada conector debe gestionar los espacios de nombre. Los conectores File System y SharePoint usan el nombre *adaptor.namespace* como entrada de configuración.

- No existe el concepto de espacio de nombres local, así que puede especificar el espacio de nombres que desee. Las ACL de estos conectores usan el mismo espacio de nombres, salvo en este caso:
    - Los conectores 4.0 ya no usan **principal-type**. El ámbito de los grupos de SharePoint se añaden al espacio de nombres y los objetos principales se envían sin el prefijo. Por ejemplo, "My SP Group" de http://sharepointhost/sitecollection/ lo procesará el conector SharePoint de la forma siguiente (suponiendo que el grupo de credenciales está definido con la opción predeterminada):

      *Espacio de nombres: Default_*http://sharepointhost/sitecollection/
      Nombre del objeto principal: My SP Group

      Si el objeto principal tiene un dominio como, por ejemplo, mydomain\mygroup, se procesará de esta manera:

      Espacio de nombres: Default
      Nombre del objeto principal: mygroup
      Dominio: mydomain

Autenticación

Tal y como se explica en el Capítulo 1, la autenticación CONNECTOR usa el protocolo SAML. La estructura para conectores 4.0 proporciona SAML como base para la seguridad. Los conectores que se basan en la nueva estructura deben proporcionar una implementación propia del proceso de autenticación para la fuente de contenido en cuestión. A continuación, le indicamos la configuración que debe aplicar en la Consola del administrador. En **Búsqueda > Búsqueda segura > Mecanismos de autenticación de acceso universal > SAML**, introduzca los valores siguientes:

ID de identidad de IDP: la entrada de configuración **server.samlEntityId** del archivo de configuración del conector.
URL de inicio de sesión: https://connector-host-name:port/samlip
Clave pública: <clave pública del IdP>

Tenga en cuenta la información siguiente sobre la implementación SAML por parte del conector:

- Puede proporcionar autenticación más de un conector. Los ID de identidad serán diferentes.
- Solo se admite el enlace en tiempo de compilación.
- El extremo del IdP SAML "samlip" está codificado.
- Se pueden devolver grupos como parte de la confirmación SAML en el atributo "member-of".

Autorización

En esta sección el concepto "autorización" hace referencia al enlace en tiempo de ejecución cuando se usa un conector 4.0. Para configurarlo, siga estos pasos: en la Consola del administrador, en **Búsqueda > Búsqueda segura > Autorización flexible**, la **URL de servicio de autorización** debe definirse como: https://connector-host-name:port/saml-authz.

Seguridad en entornos Windows

La mayoría de las implantaciones del dispositivo, que incorporan búsqueda de contenido protegido, ocurren en un entorno Microsoft Windows. Google proporciona dos productos complementarios para la integración: SAML Bridge y el conector Active Directory Groups.

SAML Bridge

El dispositivo de búsqueda admite la autenticación Kerberos directamente en Windows sin necesidad de instalar componentes externos a GSA. Puesto que Kerberos es compatible en todos los entornos Windows, es el mecanismo recomendado para la autenticación silenciosa. Sin embargo, puede resultar insuficiente por los motivos siguientes:

1. Kerberos es un sistema bastante sensible al entorno. Por ejemplo, es posible que un dispositivo cliente no admita Kerberos o que un entorno de red no sea compatible con Kerberos. En esos casos, los clientes Windows nativos usan como alternativa la autenticación NTLM. Sin embargo, el dispositivo de búsqueda no admite NTLM de forma nativa, así que no cuenta con ningún mecanismo alternativo.

2. En algunas empresas no se permite usar archivos de tabla de claves para Kerberos. GSA emplea un archivo de este tipo para habilitar Kerberos.

3. Cuando GSA está habilitado para Kerberos y se usa para autorizaciones de solicitud HEAD, solo puede realizar delegaciones sin restricciones. Algunas empresas no lo admiten.

Si desea habilitar la autenticación silenciosa cuando no se pueda usar Kerberos (o el archivo de tabla de claves), deberá configurar un proceso de autenticación externo. Google proporciona una herramienta de código abierto denominada SAML Bridge compatible con estas situaciones. Se trata de una solución basada en SAML que se ejecuta en la infraestructura Windows, por lo que debe instalarse en un host aparte; permite autenticar a usuarios mediante NTLM o Kerberos. Si desea obtener información detallada sobre la configuración de SAML Bridge, consulte el documento sobre cómo habilitar la autenticación integrada en Windows.

Conector Active Directory Groups

En un entorno Windows, muchas fuentes de contenido están integradas en Active Directory. Los grupos de Active Directory se usan para controlar el acceso a ciertos recursos. El conector Active Directory Groups de Google Search Appliance es una herramienta que puede servir para el enlace en tiempo de compilación. Es la opción preferida para resolver grupos que se necesita para el enlace en tiempo de compilación frente a la autenticación LDAP, que se puede configurar directamente en GSA. Si bien la autenticación LDAP también puede usarse para resolver grupos de Active Directory, es en el enlace en tiempo de ejecución donde se resuelven; durante el proceso de autenticación, el dispositivo intenta comunicarse con controladores de dominio directamente para obtener los grupos asociados a un determinado usuario. Como alternativa, el conector Active Directory Groups realiza el enlace en tiempo de compilación de la resolución de grupos. Realiza la transferencia a Active Directory y almacena la información sobre la pertenencia a grupos de los usuarios en su propia base de datos. Al mostrar los resultados, el conector lee de esta base de datos en lugar de comunicarse directamente con los controladores de dominio. Es mucho más eficaz, sobre todo
en un entorno a gran escala con varios dominios.

A continuación, se incluyen algunos comportamientos específicos y prácticas recomendadas para la implementación:

- El conector se ejecuta durante mucho tiempo. Si Active Directory incluye un gran número de usuarios y de grupos, podría ejecutarse durante días, incluso. Se recomienda lo siguiente:
    - Utilice instancias específicas del conector Active Directory Groups. Debe hacerse también incluso con el conector SharePoint, que integra capacidad para el conector Active Directory Groups y puede indexar tanto contenido de SharePoint como grupos de Active Directory.
    - Aumente el tiempo de espera de la transferencia. La transferencia consta de seis fases, que puede comprobar a través de los registros. Si aparece "update 1/6" y "update 2/6" de forma repetida, pero no hay más información posterior, significa que se ha interrumpido el subproceso de la transferencia y, por lo tanto, no se ha completado. Para aumentar el tiempo, cambie la variable traversal.time.limit en INSTALLROOT/INSTANCENAME/ Tomcat/webapps/connector-manager/WEB-INF/applicationContext.properties.
- Asegúrese de que el enlace se hace directamente a un host de controlador de dominios con desequilibrio de carga para aprovechar la transferencia en incrementos de Active Directory.
    - El conector utiliza el punto de comprobación exclusivo de un determinado controlador de dominios. Por lo tanto, para poder beneficiarse de las actualizaciones del punto de comprobación, debe seguir conectado al mismo controlador de dominios único en cada solicitud.
- Utilice siempre un conector no integrado y una instancia de conector por cada gestor de conector.
    - Es más fácil aplicar las revisiones y solucionar los problemas.
    - Es más escalable porque puede controlar el consumo de recursos fácilmente.
- Utilice una base de datos externa para almacenar la información del grupo.
    - Es más fiable para cuestiones de producción que usar la base de datos integrada.
    - Dado que la base de datos integrada está enlazada a una instancia de gestor de conectores, es también la única forma de resolver grupos correctamente cuando se usan varias combinaciones de conectores Active Directory Groups y SharePoint con varios gestores de conectores. Por ejemplo, si hay varios dominios de Active Directory, debe haber un conector para cada dominio. Para poder resolver grupos de usuarios de distintos dominios o si los usuarios pertenecen a grupos de varios dominios, la información de los grupos debe incluirse en la misma base de datos y tablas. Dado que la configuración de la base de datos se encuentra en el nivel del gestor de conectores, deberá configurar dichos gestores para que usen una base de datos externa y que las diferentes instancias compartan los mismos datos relacionados.

Seguridad del perímetro

Los documentos que se incluyen en el índice del dispositivo de búsqueda pueden etiquetarse como públicos ("public") o como protegidos ("secure"). Esta etiqueta dependerá de cómo se haya indexado el contenido, ya sea por rastreo o por feeds, y de la información de configuración de GSA. En términos de seguridad, un documento indexado pertenece a una de estas dos categorías:

Documento público	Documento protegido
• Documento público rastreado • Documento de feed sin protección • Contenido de una fuente de contenido protegido que se ha marcado como pública a través de la Consola del administrador de GSA	• Documento rastreado de forma segura • Documento de feed que se ha declarado protegido

Los usuarios pueden realizar búsquedas y acceder a documentos públicos sin necesidad de autenticarse. Sin embargo, hay una excepción. Con GSA 6.14 se introdujo la función de seguridad del perímetro
para GSA, por la que se garantiza que el dispositivo de búsqueda no muestra ningún resultado sin que se autentique al usuario. Cuando se habilita la función de seguridad del perímetro, el dispositivo de búsqueda debe autenticar al usuario mediante uno de los mecanismos de autenticación configurados antes de mostrar cualquier resultado. Si la autenticación no se realiza correctamente, GSA no muestra los resultados, aunque sean públicos. Tenga en cuenta que la autenticación se realiza para los documentos marcados como públicos, sin necesidad de procesar autorización alguna.

Para habilitar la seguridad del perímetro, deberá configurar un mecanismo de autenticación; puede ser cualquiera de los que se describen en el Capítulo 2. A continuación, vaya a **Publicación -> Acceso universal** y habilite la seguridad del perímetro. Una vez que la seguridad del perímetro esté habilitada, se aplica a GSA de forma global y no se puede configurar por colección ni por servidor.

Ejemplo de búsqueda segura

A continuación, se indican los requisitos de cuatro fuentes de contenido que deben incluirse en la búsqueda (todas ellas protegidas):

1. SharePoint 2010 con autenticación Kerberos. Para indexar el contenido se utiliza el conector SharePoint compatible con Google.
2. Contenido de Salesforce integrado en un IdP SAML que emplea la autenticación basada en formularios, pero donde el directorio de usuarios sigue siendo Active Directory. Se implanta un conector Salesforce para indexar el contenido con ACLs. El conector se ha diseñado según la estructura para conectores de Google y envía documentos que empiezan por "googleconnector://".
3. Un sitio web IIS personalizado con autenticación Kerberos. No hay ninguna API para comprobar permisos o para obtener ACLs. GSA rastrea el contenido directamente.

4. Una aplicación empresarial heredada. Los usuarios y los permisos se almacenan en la base de datos; no se integran en Active Directory. No se produce una asignación directa de los nombres de usuario entre Active Directory y esta aplicación. El conector para bases de datos de Google se usa para indexar el contenido. Se puede emplear una declaración de consulta SQL para determinar si un usuario tiene acceso a los registros de la base de datos de los resultados de búsqueda.

Asimismo, hay que tener en cuenta que la empresa cuenta con varios dispositivos. Algunos no son compatibles con Kerberos.

Identidades de usuario
SharePoint, Salesforce y el sitio web IIS personalizado cuentan con el mismo Active Directory, mientras que la aplicación heredada tiene el suyo propio. Eso significa que se necesitan dos grupos de credenciales: se puede usar el grupo de credenciales predeterminado para Active Directory y añadir un grupo de credenciales heredado para la aplicación empresarial.

Autorización
Al intentar dar con una solución, hay que empezar por la autorización. Obviamente, habrá que usar ACL por URL para el contenido de Salesforce y SharePoint. Dado que el conector de GSA para bases de datos admite la autorización mediante una consulta, se puede usar la autorización CONNECTOR para este
tipo de contenido. Habrá que usar la solicitud HEAD para el sitio web IIS personalizado. Dado que utiliza Kerberos, se puede usar la solicitud HEAD con Kerberos. SharePoint, Salesforce y las aplicaciones heredadas necesitan una identidad de usuario verificada; el sitio web IIS personalizado, no.

Autenticación
Ahora que hemos decidido los mecanismos de autorización que se van a usar, hay que elegir el método de autenticación para cada grupo de credenciales. Para el grupo de credenciales predeterminado, no se puede usar Kerberos en GSA porque hay dispositivos cliente que no admiten Kerberos. Solo nos queda SAML Bridge. Si analizamos la situación más en profundidad, es posible que se pueda usar el IdP SAML disponible que utiliza la integración Salesforce. Devolverá la misma identidad verificada y no requiere otro servidor para alojar SAML Bridge.

A continuación, hay que comprobar si esta estrategia de autenticación es suficiente para los requisitos de la autorización que exige el grupo de credenciales predeterminado. Debería abarcar el contenido de Salesforce y SharePoint, puesto que se obtiene una identidad verificada que se usará para las comprobaciones de ACL. Para el sitio web IIS personalizado supone un problema usar el IdP SAML para Salesforce con autenticación de cookies, ya que no habrá incidencia de Kerberos disponible para la autorización mediante solicitud HEAD. Para cumplir este requisito, se puede usar SAML Bridge para la autorización únicamente porque admite la delegación Kerberos. Puede realizar solicitudes HEAD por lotes mediante Kerberos a partir de un nombre de usuario. Pero eso significa que aun así hay que implantar SAML Bridge. Si hay que implementar SAML Bridge de todas formas, se puede usar para la autenticación también.

Para el grupo de credenciales heredado, hay que realizar la autenticación con las credenciales de usuario almacenadas en la base de datos. Sin embargo, el conector de GSA para bases de datos no proporciona ningún mecanismo de autenticación. En ese caso, hay que implantar de forma personalizada la interfaz del gestor de autenticación del gestor de conectores en el conector para bases de datos.

Ahora que ya sabemos qué mecanismos de autenticación se van a usar, hay que configurar las dos reglas siguientes en "Mecanismos de autenticación de acceso universal":

1. **SAML**. Si se utiliza el grupo de credenciales predeterminado, hay que configurar SAML Bridge en el modo de enlace POST. Para obtener instrucciones más detalladas al respecto, consulte este documento de la wiki.

2. **CONNECTOR**. Si se usa el grupo de credenciales heredado, hay que configurar el conector personalizado para la base de datos.

Reglas de la autorización flexible
En general, para la mayoría de las implantaciones, se pueden dejar tal cual las tres primeras entradas de la autorización flexible: PER_URL_ACL, CACHE y POLICY. Esto también es válido para esta implementación en particular. La regla ACL por URL se ejecutará para contenido Salesforce y SharePoint porque las ACL se indexan con documentos. Hay que hacer algunos cambios en la regla CONNECTOR porque la configuración predeterminada solo está asociada con el grupo de credenciales predeterminado.

- CONNECTOR
    - Cambie **ID de autenticación** a "heredado"; aquí equivale a la selección del grupo de credenciales.
    - Rellene el nombre del conector para la base de datos en el campo **Nombre de conector**.

También hay que definir una regla SAML. Aunque SAML Bridge utiliza las solicitudes HEAD para autorizar sitios web IIS personalizados, no podemos confiar en la regla HEADREQUEST porque la usa GSA para realizar solicitudes HEAD.

- SAML
    - Debería aparecer justo después de la regla CONNECTOR en la autorización flexible.
    - **ID de autenticación** debería estar definido con la opción predeterminada (se asigna al grupo de credenciales).

URL del servicio de autorización debería apuntar a Authz.aspx. de SAML Bridge.

Capítulo 3 Autenticación para desarrolladores

Siempre que sea posible en sus implementaciones, debería usar productos existentes, compatibles con Google, proporcionados por partners de Google o productos estándar de terceros. En general, si sigue las directrices de este documento conseguirá minimizar los riesgos del proyecto y reducir los costes globales de propiedad. Sin embargo, es posible que para cumplir algunos requisitos deba desarrollar aplicaciones o procesos personalizados externos con el fin de implementar por completo la seguridad o la integración del contenido en GSA.

GSA ofrece las opciones siguientes para crear procesos de autenticación personalizados externos:

- Autenticación basada en formularios con acceso a cookies
- SAML
- Estructura para conectores
- Aplicación de confianza*¡Nuevo!*

Autenticación basada en formularios con acceso a cookies

Si los sistemas ya emplean la autenticación basada en cookies, una opción para integrar seguridad en GSA es reutilizar y personalizar el proceso de autenticación que ya existe para crear un servicio de autenticación silencioso en GSA. El acceso a cookies es una personalización posible del proceso de autenticación basada en formularios existente que permite extraer la identidad del usuario a partir de la cookie de autenticación de este y enviarla al dispositivo de búsqueda.

El proceso de acceso a cookies debe descubrir quién es el usuario que está usando la cookie comunicándose con una URL externa, utilizando APIs de inicio de sesión único o similares y enviando credenciales del usuario como cabeceras HTTP al dispositivo de búsqueda de forma segura. Antes de llegar al dispositivo de búsqueda, el sistema de inicio de sesión único que ha creado la cookie de la sesión debe haber autenticado al usuario. De lo contrario, se redirige al usuario a una página de inicio de sesión para definir la identidad en el sistema de inicio de sesión único. Posteriormente, las credenciales se envían a GSA.

Para implementar el acceso a cookies con su sistema basado en formularios o de inicio de sesión único, deberá configurar una URL externa protegida por el sistema de inicio de sesión único. Durante el proceso de autenticación, GSA se comunica con esa URL y reenvía las cookies de la sesión que ya ha creado
el sistema de inicio de sesión único. De esta manera, el servicio externo puede verificar la identidad del usuario y enviarla de nuevo al dispositivo de búsqueda en una cabecera HTTP denominada `X-Username` y, de forma opcional, `X-Groups`. Este proceso se puede personalizar por completo y crear un modelo
de la seguridad para su proyecto de búsqueda empresarial.

Para crear el proceso de acceso a cookies, siga estos pasos:

1. Cree una aplicación web que pueda validar la identidad de un usuario a partir de una cookie de sesión de un sistema de inicio de sesión único.
2. Configure una regla de autenticación basada en formularios en la Consola del administrador de GSA.

Consideraciones importantes
Si desea aplicar un servicio de autenticación silenciosa con su sistema de inicio de sesión único, tenga en cuenta lo siguiente:

- Las cookies de sesión no deben estar limitadas a un mismo IP de usuario; algunos sistemas de inicio de sesión único ofrecen esta restricción como una medida de seguridad.

- GSA debe formar parte del mismo dominio que la cookie de sesión que utiliza el sistema de inicio de sesión único. Por ejemplo, si una cookie utiliza el dominio "foo.com", hay que configurar GSA como parte de ese dominio, es decir, "gsa.foo.com". De esta manera, el navegador enviará la cookie del sistema de inicio de sesión único a GSA. El dominio de la cookie y de GSA debe asociarse y sincronizarse correctamente.

- La aplicación de acceso a cookies debe ser una aplicación web simple escrita en cualquier lenguaje de programación que admita su servidor web o de aplicaciones como, por ejemplo, Java o .NET. La aplicación envía de nuevo el nombre de usuario al dispositivo de búsqueda a través de una cabecera HTTP personalizada. Por ejemplo:
  `X-Username: luis.sanchez`

- La aplicación de acceso a cookies se suele implementar detrás del sistema de inicio de sesión único. Es decir, se puede instalar detrás del plug-in web de inicio de sesión único que autentica al usuario y normalmente transfiere esa identidad a aplicaciones web de confianza en una cabecera HTTP de forma segura.

- No tiene que ser una aplicación independiente. Puede ser otra página ASP, JSP u otra página web dinámica en una aplicación existente protegida por el mismo formulario de inicio de sesión.

- El proceso de acceso a cookies no significa que se acceda realmente a las cookies, sino que, después de la autenticación, se genera una cookie que indica una sesión de usuario válida. El nombre de usuario se puede obtener a través de varios métodos. Por ejemplo, un sistema de inicio de sesión único comercial como SiteMinder utiliza la cabecera HTTP_SM_USER para añadir el ID de usuario.

- Si no es posible usar un plug-in web de inicio de sesión único para esta aplicación web personalizada y facilitar la implementación de la solución, puede usar el API de inicio de sesión único para extraer el usuario asociado a la cookie. Nunca manipule un sistema de inicio de sesión único en producción para que transfiera el ID de usuario de una cookie en texto sin formato. Supone un riesgo importante para la seguridad.

- Tenga en cuenta que algunos sistemas de inicio de sesión único se pueden configurar para transferir esas credenciales en una cabecera HTTP sin tener que desarrollar una aplicación web.

Resolución de grupos del enlace en tiempo de compilación (autorización ACL)
La respuesta de autenticación del proceso de acceso a cookies también se puede ampliar para incluir información de grupo del usuario que se autentica. El proceso de acceso a cookies no variaría. El único cambio necesario serían las cabeceras de la respuesta, que ahora también contendrían información de grupo del usuario en cuestión. Los grupos que resuelva el mecanismo de autenticación de cookies pertenecerán al espacio de nombres global del grupo de credenciales que se haya seleccionado para el mecanismo. Si se necesitan grupos para la sesión autenticada, se añadiría una cabecera HTTP con X-Groups:

```
X-Groups: department_users, enterprise_users
```

SAML

El dispositivo de búsqueda admite SAML 2.0, un protocolo basado en XML para un proveedor de identidades externo. Habrá casos en los que quizá deba desarrollar un IdP SAML personalizado. Tenga en cuenta que crear un IdP SAML desde cero lleva bastante tiempo. Debería empezar con una base de código existente, como OpenSAML. Google también ofrece un proyecto de código abierto, SAML Bridge, para autenticaciones silenciosas con tecnologías Windows.

Tenga en cuenta que hay dos formas distintas de configurar la autenticación SAML en el dispositivo de búsqueda:

- Enlace de artefacto de HTTP. El navegador es el mecanismo de comunicación principal entre GSA (proveedor de servicios) y el proveedor de identidades (su servidor SAML).
- Enlace POST de HTTP. Requiere un mecanismo de confianza entre GSA (proveedor de servicios) y el proveedor de identidades (su servidor SAML).

El objetivo de este documento no es ofrecer directrices para esta configuración; esto ya se describe en la documentación. Sin embargo, en la tabla siguiente encontrará consejos sobre qué enlace SAML usar.

	Enlace de artefacto de HTTP SAML	Enlace POST de HTTP SAML
Requisitos	Se requieren varias redirecciones transparentes. Se requiere confianza entre el navegador y GSA/proveedor de identidades SAML.	Además de la confianza entre el navegador y los demás servidores, deberá crear un enlace de confianza entre GSA y el proveedor de servicios.
Configuración	Se recomiendan conexiones HTTP seguras del cliente hacia GSA y el proveedor de identidades. Hay que configurar otra URL en GSA (URL de resolución de artefactos) para poder ofrecer información de autenticación definitiva a GSA.	Además de tener conexiones SSL entre el cliente y los servidores, también exige configurar certificados entre GSA y el proveedor de identidades. Se utiliza para que GSA obtenga la información de autenticación directamente del proveedor de servicios.
Infraestructura de clave pública	Es menos complejo desde la perspectiva de la seguridad porque no exige una solución PKI (Public Key Infrastructure) o infraestructura de clave pública.	Deben emitirse certificados de confianza; por lo tanto, se necesita una infraestructura PKI. Tenga en cuenta que también se puede hacer a través de una solución OpenSSL.
Alta disponibilidad	Es más complejo proporcionar una solución con una gran disponibilidad para GSA y el proveedor de identidades debido a las distintas redirecciones de navegador y la persistencia que exige el artefacto entre las distintas invocaciones.	Se puede implementar una solución con una gran disponibilidad.
Proveedor de identidades SAML	Si tiene que desarrollar un proveedor de identidades SAML basado en enlaces de artefactos desde cero, sería más complejo porque exige un servicio adicional (URL de resolución de artefactos). En Internet encontrará algunas estructuras de código abierto, como OpenSAML, y muchos ejemplos de código.	Quizá sería más sencillo desarrollar un proveedor de identidades SAML desde cero, pero deben gestionarse las firmas digitales en su código.

En general, es preferible usar enlaces POST de HTTP SAML porque ofrecen una solución más potente y simple, principalmente en términos de alta disponibilidad.

Resolución de grupos del enlace en tiempo de compilación (autorización ACL)
La respuesta de autenticación SAML se puede ampliar para incluir grupos de usuarios con el fin de devolver la autenticación de usuario a GSA. Los grupos que resuelva el mecanismo de autenticación SAML pertenecerán al espacio de nombres global del grupo de credenciales que se haya seleccionado para el mecanismo.

Ejemplo de respuesta SAML:
En ese ejemplo, la respuesta SAML incluye tanto el nombre de usuario ("Subject") como una declaración de atributo AttributeStatement "member-of" con los grupos del usuario resueltos.

```
<Assertion Version="2.0"
        ID="blahblah2"
        IssueInstant="2011-01-01T14:38:05Z"
        xmlns="urn:oasis:names:tc:SAML:2.0:assertion">
  <Issuer>ac.corp.company.com</Issuer>
  <Subject>
    <NameID>luis.sanchez</NameID>
  </Subject>
  <AuthnStatement AuthnInstant="20011-01-01 T14:38:05Z">
    <AuthnContext>
      <AuthnContextClassRef>
        urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
      </AuthnContextClassRef>
    </AuthnContext>
  </AuthnStatement>
  <AttributeStatement>
    <Attribute Name="member-of">
      <AttributeValue>marketing</AttributeValue>
      <AttributeValue>us-employees</AttributeValue>
      <AttributeValue>SFO-office</AttributeValue>
    </Attribute>
  </AttributeStatement>
</Assertion>
```

Acceso a cookies y SAML

Si debe personalizar el proceso de autenticación, es importante diferenciar entre el acceso a cookies y SAML para diseñar el mejor plan posible antes de empezar con el proyecto.

	SAML	Acceso a cookies
Integración	Algunos sistemas de inicio de sesión único ofrecen una interfaz de autenticación SAML que puede integrarse en el dispositivo sin necesidad de configurarla.	Algunos sistemas de inicio de sesión único pueden integrarse fácilmente a través del proceso de acceso a cookies.
Complejidad	Sería más complejo si tuviera que desarrollar un proveedor SAML desde cero.	Los costes de desarrollo de una solución de acceso a cookies para el dispositivo podrían ser menores.
Autenticación	Hay una interacción entre el navegador (usuario) y el proveedor de servicios, por lo que se puede usar con cualquier protocolo de autenticación directo, como Kerberos o NTLM.	En este proceso de autenticación, el dispositivo se comunica con la URL de muestra sin interacción por parte del usuario. Por lo tanto, solo es válido para un plan de autenticación basado en cookies.

Si desea obtener los detalles técnicos exactos sobre cómo implementar ambos sistemas, consulte los apartados sobre el acceso a cookies y SPI de autenticación y de autorización. Es importante comprender bien los flujos de las interacciones de ambos para así implantar los procesos correctamente.

Estructura para conectores para la resolución de grupos

La estructura para conectores también ofrece una interfaz para autenticar a los usuarios. Sin embargo, puesto que no se trata de un mecanismo de autenticación silenciosa, no se recomienda la autenticación CONNECTOR. Por otro lado, el conector se puede implementar para proporcionar la resolución de grupos en el enlace en tiempo de compilación, lo que resulta mucho más útil. Es habitual que un mecanismo de autenticación silenciosa, como Kerberos, SAML o el acceso a cookies se combine con una resolución
de grupos basada en un conector.

Compatibilidad de interfaz
En la estructura para conectores se define la interfaz siguiente, que deberá implementar un desarrollador de conectores:

```
public AuthenticationResponse authenticate(final AuthenticationIdentity
identity)
      throws RepositoryLoginException, RepositoryException
```

Cuando se configura un conector en "Mecanismos de autenticación de acceso universal", se ofrece la opción para procesar resoluciones de grupos únicamente.

Cuando el conector deba proporcionar tanto la autenticación como la resolución de grupos, la implementación puede ignorar lo que GSA le transfiere a través del objeto AuthenticationIdentity y devolver grupos y un nombre de usuario verificados a GSA a través de AuthenticationResponse. Este es el constructor de AuthenticationResponse:

```
public AuthenticationResponse(boolean valid, String data,Collection<?> groups);
```

"valid" indica si la autenticación se ha realizado correctamente o no, en caso de que se haya utilizado el método "authenticate()" para llevar a cabo la autenticación. "data" se usará posteriormente. La colección "groups" se usará para incluir los grupos de usuarios de la clase siguiente:

```
public class Principal;
```

Esta clase incluye información sobre los grupos: nombre, espacios de nombres, distinción entre mayúsculas y minúsculas, y el atributo principal_type. Son atributos de una ACL por URL. El último atributo principle_type lo utiliza la infraestructura para conectores con el fin de introducir el concepto de "espacio de nombres local". Consulte la sección sobre la descripción general de espacios de nombres para obtener más información acerca de cómo asignar los espacios de nombres. A continuación se explica cómo se usa durante la transferencia:

1. El conector se configura en la Consola del administrador de GSA con un espacio de nombres global (grupos de credenciales) y con un espacio de nombres local.
2. El conector recopila todos los grupos y crea el objeto **principal**. Si son locales en una determinada fuente de contenido, principal_type se define en "unqualified". El nombre del espacio de nombres local se añade como prefijo al nombre del grupo como si fuera un dominio.
3. El gestor de conectores traduce los objetos **principal** según la definición de ACL en feeds XML con las propiedades de los objetos **principal** a los atributos de ACL correspondientes.

Compatibilidad de base de datos
Los conectores pueden resolver grupos al mostrar los resultados a partir de estas dos opciones de diseño:

1. El conector puede consultar la aplicación en la que se almacena la información de pertenencia a grupos al mostrar los resultados si hay un API disponible. El conector Documentum (versión 3.2), compatible con Google y que admite el enlace en tiempo de compilación, utiliza este proceso.
2. Se puede obtener la información de los usuarios, de los grupos y de las relaciones entre estos de antemano y guardarse en el espacio de almacenamiento del propio conector. Al mostrar los resultados, el conector lee los datos de este espacio de almacenamiento y los envía al dispositivo. Ejemplo de ello son los conectores Active Directory Groups y SharePoint que proporciona Google. A veces, es la única opción. Por ejemplo, si hay varios dominios de Active Directory, realizar
   la consulta desde todos ellos en el momento de mostrar los resultados puede ralentizar considerablemente el proceso y, por lo tanto, la invocación en tiempo real no sería posible.

La estructura para conectores incluye compatibilidad con bases de datos. El archivo de configuración applicationContext.properties incluye ajustes de configuración JDBC para distintas bases de datos. La estructura para conectores integra una base de datos H2. Si desarrolla un conector, podrá almacenar la información sobre la pertenencia a grupos de los usuarios en la base de datos mediante el segundo proceso descrito anteriormente.

Aplicación de confianza(beta)

Es muy habitual que GSA se implemente en un portal para así ofrecer el servicio de búsqueda deseado. La interfaz de búsqueda se incluye en el portal, por lo que los usuarios no interactúan directamente con el dispositivo. La dificultad que se plantea en un entorno de este tipo es cómo transferir las credenciales del usuario a GSA sin pedirle que inicie sesión. Para conseguirlo, la página de búsqueda debe simular el comportamiento de un navegador mientras interactúa con el dispositivo, obtener las credenciales del usuario de la sesión del portal y transferir esos datos al dispositivo. Podría ser bastante complicado teniendo en cuenta la variedad de protocolos de autenticación que usan las aplicaciones de portales y la forma en que se gestionan las credenciales del usuario final.

Para simplificarles el proceso de integración a los desarrolladores de portales, Google Search Appliance ha añadido una función nueva en la versión 7.2: aplicación de confianza. El concepto es simple: los portales utilizan una cuenta de usuario de confianza preconfigurada para establecer la sesión con GSA mediante protocolos de autenticación limitados pero simples y envía solicitudes de búsqueda segura con el nombre de usuario del usuario final. Evita los problemas mencionados anteriormente con lo siguiente:

1. Los protocolos de autenticación simples son fáciles de gestionar. Se admiten dos mecanismos: la autenticación básica y el acceso a cookies.
2. Puesto que solo se requieren nombres de usuario, no supone ningún problema el hecho de no poder obtener la contraseña del usuario en la sesión iniciada en el portal.

Las identidades de los usuarios finales se envían a través de las dos cabeceras HTTP personalizadas siguientes:

X_GSA_USER: cabecera que incluye el nombre de usuario
X_GSA_CREDENTIAL_GROUP: cabecera que identifica el grupo de credenciales del usuario final.

Consideraciones importantes

1. Cuando se realice una búsqueda segura se activarán todos los mecanismos de autenticación que se hayan configurado, incluido el mecanismo configurado para la aplicación de confianza.
2. Puede usar cualquier lenguaje de programación: Python, Java, C#.
3. Es necesario verificar al usuario de confianza. Dado que GSA admite el acceso a cookies y la autenticación básica, el rendimiento se verá afectado por el servidor de contenido. Debería evitar la autenticación siempre que sea posible.

4. Puede usar la cookie de sesión de GSA que se ha devuelto de una invocación a GSA para invocaciones posteriores.

5. Todas las solicitudes de búsqueda deben ir acompañadas del nombre de usuario y del grupo de credenciales para poder ofrecer los resultados al usuario adecuado.

6. Durante el proceso de autenticación se activará la resolución de grupos del usuario final. Si hay una resolución de grupos configurada, se invocará. Dado que la resolución de grupos también tiene consecuencias para el rendimiento, debe evitar el proceso para cada una de las invocaciones del API.

7. Si se vuelve a enviar el mismo usuario final para otra búsqueda con la misma cookie de sesión válida de GSA, la resolución de grupos no se activa de nuevo. Si se envía un usuario final distinto por primera vez, aunque le acompañe la misma cookie de sesión de GSA, la resolución de grupos se activa para dicho usuario.

8. Cuando caduca la sesión del usuario de confianza (la caducidad de las cookies depende del ajuste de tiempo de espera de la sesión que se encuentra en **Búsqueda segura -> Control de acceso**), GSA devuelve un error: "El servidor remoto ha devuelto un error: (502) pasarela incorrecta".

9. Cuando la sesión del usuario de confianza es válida (es decir, no ha superado el valor del tiempo de espera de la sesión), pero caduca la duración de la confianza en el mecanismo de autenticación, el dispositivo realiza otra autenticación con las credenciales del usuario de confianza. La invocación se ralentiza tanto como la primera invocación que ha devuelto la cookie de sesión de GSA actual.

Prácticas recomendadas

1. Si se trata de una integración con un portal, la cookie de sesión de GSA que se devuelve debería almacenarse en la sesión activa del usuario en el portal. Esto significa que cada usuario del portal tendrá una sesión de GSA propia y que deben almacenarse todas. También significa que estas cookies de sesión de GSA se reutilizarán para el mismo usuario final y así ahorrar costes con la autenticación de usuarios de confianza.

2. En cuanto al rendimiento, es preferible que la sesión de GSA permanezca válida durante la sesión del usuario en el portal. Sin embargo, no hay garantía de que esto ocurra, ya que el usuario puede navegar por el portal un buen rato antes de volver a hacer otra búsqueda. En cuanto al código, deberá gestionar el hecho de que se produzca otra invocación una vez que la cookie de sesión
de GSA haya caducado.

3. Establezca la duración de confianza del mecanismo de autenticación en el mismo valor que el tiempo de espera de la sesión. El valor predeterminado es 1.800 segundos. Si lo hace, evitará el impacto en el rendimiento de otra autenticación implícita que utiliza la credencial del usuario de confianza.

4. El nombre de dominio debería transferirse como prefijo en los nombres de usuario de los usuarios finales. De lo contrario, se producirá un error en la invocación.

Consulte un cliente de muestra en C# en el Apéndice A. Debe realizarse una instancia de la clase por cada sesión de usuario en el portal y se vuelve a intentar una vez cuando la sesión de GSA caduca.

Autenticación de los conectores 4.0 (beta)

Un conector solo debe proporcionar implementación para la interfaz siguiente:

public interface AuthnAuthority

y registrarla en:

AdaptorContext.setAuthnAuthority()

Si desea obtener información sobre la implementación, consulte el adaptador de autenticación de Google. Después de la autenticación, se devuelve un objeto de clase **AuthnIdentity**. Incluye el nombre de usuario y, de forma opcional, grupos o contraseñas.

Capítulo 4 Autorización para desarrolladores

Descripción general

Un motor de búsqueda empresarial debe mostrar resultados relevantes al usuario, pero solo aquellos resultados a los que tenga acceso. Esto se gestiona a través del proceso de autorización que afecta a todos los documentos protegidos del índice. En este capítulo nos centraremos en las soluciones personalizadas a la hora de diseñar el proceso de autorización dentro de su proyecto de búsqueda empresarial con Google.

En la sección sobre la selección de un proceso de autorización se han expuesto las siguientes opciones más importantes para crear un proceso de autorización personalizado:

- ACLs por URL
- Políticas ACL
- Autorización SAML
- Conectores

En los apartados siguientes se incluye información más detallada sobre cómo usar estas opciones en una solución personalizada.

ACLs por URL

El mayor reto a la hora de usar enlaces en el tiempo de ejecución con un conector personalizado o con feeds es simular el modelo de autorización del sistema objetivo. Cada sistema puede tener un modelo de seguridad diferente.

Se puede asociar las ACL a los documentos de dos formas: como metadatos en las cabeceras HTML o a través de cabeceras HTTP personalizadas. Sin embargo, solo los feeds permiten especificar todos los atributos de ACL posibles. Dado que la estructura para conectores de Google se basa en feeds, esto afecta también a los casos en los que las ACL se envían a través de un conector. Si desea obtener información sobre cómo definir la ACL en su totalidad, consulte la sección acerca de cómo especificar ACLs por URL. Entre las funciones que GSA ofrece para simular distintos modelos de seguridad, la herencia de ACLs es muy importante. Gracias a la

herencia de ACLs, gestionar los cambios en las ACL resulta mucho más eficaz. Dado que las ACL ya no tienen que ampliarse ni adjuntarse a cada uno de los niveles jerárquicos, gestionar los cambios en las ACL es mucho más eficiente; solo hay que volver a indexar el nivel en el que se ha realizado el cambio de permiso.

El atributo "inheritance-type" permite crear un modelo de los distintos mecanismos de seguridad de los diferentes sistemas de contenido. En una cadena de herencias, el proceso de comprobación de permisos siempre vuelve al principio y los permisos se evalúan en función del tipo de herencia que se haya definido:

- PARENT_OVERRIDES
    - El permiso de la ACL principal prevalece sobre el permiso de la ACL secundaria, salvo cuando el permiso principal es INDETERMINATE. En ese caso, prevalece el permiso secundario. Si el valor tanto de la ACL principal como de la ACL secundaria es INDETERMINATE, el permiso será INDETERMINATE.
- CHILD_OVERRIDES
    - El permiso de la ACL secundaria prevalece sobre el permiso de la ACL principal, salvo cuando el permiso secundario es INDETERMINATE. En ese caso, prevalece el permiso principal. Si el valor tanto de la ACL principal como de la ACL secundaria es INDETERMINATE, el permiso será INDETERMINATE.
- AND_BOTH_PERMIT
    - Este permiso será PERMIT únicamente si los permisos de la ACL principal y de la ACL secundaria son PERMIT. De lo contario, el permiso será DENY.

Ejemplo de cadena de herencia
URLs

- "FileUrl" (USER:joe access:PERMIT type:LEAF) inherits
- "FolderUrl" (GROUP:eng access:PERMIT type:CHILD_OVERRIDES) inherits
- "ShareUrl" (GROUP:interns access:DENY type:PARENT_OVERRIDES

Decisiones de autorización

- PERMITs identity (USER:joe, GROUP:eng)
    - PERMIT by FileUrl ACL, not overridden = PERMIT
- PERMITs identity (USER:moe, GROUP:eng)
    - INDETERMINATE + PERMIT + not overridden = PERMIT
- DENYs (USER:adam, GROUP:eng, GROUP:interns)
    - INDETERMINATE + PERMIT + DENY (override) = DENY

Las ACL pueden ser libres o estar enlazadas. Las ACL adjuntas a los documentos indexados están "enlazadas". Las ACL "libres" pueden representar elementos que no son del documento. Por ejemplo, algunos sistemas de contenido definen los objetos permiso, que pueden usar distintos documentos. Las ACL se mantienen en estos objetos especiales, en lugar de hacerlo en los documentos. Los sistemas de contenido, como File System, tienen jerarquías y las ACL se pueden definir en carpetas que no sean documentos. En ambos casos se pueden usar las ACL libres. No se consideran documentos indexados, por lo que no contabilizan para la licencia de GSA.

Ejemplo de ACL libre

```
<group>
  <acl url='http://dummyhost.corp.google.com/'
      inheritance-type="child-overrides" inherit-
from='http://corp.google.com/'>
    <principal scope="user" access="permit">edward</principal>
    <principal scope="user" access="deny"> william</principal>
    <principal scope="user" access="deny"> ben </principal>
    <principal scope="group" access="permit">nobles</principal>
    <principal scope="group" access="deny">playwrights</principal>
  </acl>
  ...
  ...
</group>
```

En este ejemplo, `http://dummyhost.corp.google.com/` es una ACL libre que hereda de `http://corp.google.com/` y define otros objetos principal. Dado que el tipo de herencia de la ACL es child-overrides, el objeto secundario sobrescribirá esta ACL, si existe.

Autorización SAML

Puede personalizar el proceso de autorización por completo a través de un proveedor SAML externo que resuelva autorizaciones. Lo más apropiado sería crear el proceso de autorización SAML en el lenguaje de programación que mejor conozca. La solicitud de autorización SAML es una solicitud en formato XML que el dispositivo de búsqueda envía a la URL de servicio que ha configurado en la Consola del administrador. La solicitud contiene información sobre el usuario y las URL que se deben autorizar. SAML también
admite el proceso por lotes, de modo que se pueden enviar varias URL al mismo tiempo. Dicho proceso es muy recomendable si se usa este tipo de autorización para mejorar el rendimiento y evitar un exceso de autorizaciones.

En la guía sobre autenticación/autorización SPI para empresas encontrará más información sobre el formato XML SAML, que podrá usar para crear un proceso de autorización SAML personalizado. Debe implementar el servicio que se ejecuta en un servidor de aplicaciones externo y que analiza la respuesta, extrae la información sobre si el usuario tiene permiso para acceder al documento en cuestión y devuelve una respuesta en formato XML al dispositivo de búsqueda. Ejemplo de ello es SAML Bridge, que puede realizar autorizaciones por lotes de contenido basado en Kerberos mediante solicitudes HEAD.

Consideraciones importantes
Si va a usar la autorización SAML, tenga en cuenta lo siguiente:

- La principal ventaja de implementar este modelo de autorización es que puede controlar por completo el proceso de seguridad durante la búsqueda.
- El mayor inconveniente es que está intrínsecamente relacionado con el método de enlace en el tiempo de ejecución. Por lo tanto, gestionar la autorización podría llevarle más tiempo, aunque el procesamiento por lotes podría reducirlo.

Estructura para conectores para autorizaciones

Otra opción para diseñar un modelo de la seguridad es implementar un conector personalizado. Tal y como se explica en este documento y en la documentación de GSA, se puede crear un conector para transferir o enviar por feed contenido público o protegido al dispositivo de búsqueda, así como para proporcionar autenticación y autorización al mostrar los resultados. Hemos hablado de los conectores que usan ACL por URL. En esta sección hablaremos de cómo usar los conectores para realizar una autorización como mecanismo de enlace en tiempo de ejecución.

Compatibilidad de interfaz

En la estructura para conectores se define la interfaz siguiente, que deberá implementar un desarrollador de conectores:

```
public interface AuthorizationManager;
```

Este es el método para la autorización:

```
public List authorizeDocids(Collection docids, AuthenticationIdentity
identity)throws RepositoryException;
```

"docids" es una colección de IDs de documentos únicos de los resultados de búsqueda coincidentes. Se transfieren varios "docids" del dispositivo al conector. Cuando se han autorizado suficientes documentos según la identidad del usuario de la búsqueda, el dispositivo deja de invocar al conector. De lo contario, el dispositivo sigue invocando al API (cada vez con más "docids") hasta que se agota el tiempo asignado, hasta que no hay más "docids" o hasta que se devuelven suficientes documentos con el valor PERMIT al usuario de la búsqueda.

AuthenticationIdentity contiene la identidad verificada del usuario. Según el protocolo de autenticación que se utilice, puede incluir el nombre de usuario, el dominio o incluso la contraseña (si el protocolo de autenticación implementado recaba contraseñas). Con la implementación de un conector debería definirse qué información mínima se necesita en **AuthenticationIdentity**.

Autorización de los conectores 4.0 (beta)

Un conector solo debe proporcionar implementación para la interfaz siguiente:

```
public interface AuthzAuthority
```

y registrarla en:

AdaptorContext.setAuthzAuthority()

Servidor web proxy

Las opciones descritas anteriormente son las plataformas más comunes para implementar el entorno de seguridad de la interconexión que se establece con una fuente de contenido. Pero hay otras, por ejemplo, usar un servidor web proxy para gestionar la autorización.

En este caso, la autorización se centraliza en un servidor web proxy donde todas las URL deben reescribirse para que las pueda transferir. Así pues, el dispositivo de búsqueda envía solicitudes HEAD HTTP para comprobar la seguridad antes de mostrar los resultados.

Consideraciones importantes
Utilizar un servidor web proxy es parecido a usar un proveedor de autorización SAML, pero con estos inconvenientes:

- Las solicitudes de autorización no se procesan por lotes.
- Es necesario reescribir las URL para que se puedan autorizar a través del servidor web proxy. Por ejemplo:
  `http://proxy.corp.com/proxy?returnPath=http//cont.corp.com/doc.html`
- Las URL que se han reescrito se almacenan en el índice de esta forma y quizá deban volver a traducirse a la URL original en la interfaz de búsqueda.

Resumen

En este documento se ha explicado el proceso para diseñar la seguridad de un proyecto de búsqueda empresarial con Google Search Appliance. Para ello, hay que conocer bien el área de seguridad de la empresa, así como las fuentes de contenido relacionadas que formarán parte del proyecto. Debe dedicar tiempo de calidad a analizar el caso y a diseñar un modelo de los procesos de autenticación y autorización del dispositivo de búsqueda.

Descripción general de las prácticas recomendadas sobre seguridad

- Antes de empezar con el proyecto, dedique tiempo a analizar lo siguiente:
    - ¿Qué proveedores de identidades deberá integrar para el proceso de autorización?
    - ¿Cómo autorizará los documentos de cada una de las fuentes de contenido que se encuentran integradas en GSA?

- Para integrar la seguridad en GSA, utilice siempre que sea posible componentes estándar y compatibles como, por ejemplo:
    - Kerberos
    - SAML Bridge para Windows de Google Search Appliance
    - LDAP
    - Conector Active Directory de Google Search Appliance

- Diseñe un modelo de cada uno de los proveedores de identidades que deba integrar con un grupo de credenciales.

- Clasifique los grupos de credenciales por sistemas de seguridad corporativos (proveedores de identidades) y asócielos a las fuentes de contenido correspondientes.
    - Utilice solo un grupo de credenciales por proveedor de identidades siempre que sea posible.
    - Los grupos de credenciales deberían asignarse a mecanismos de identidad únicos, y no necesariamente a fuentes de contenido.
    - Un conjunto de credenciales se puede usar en varias fuentes de contenido que compartan la misma fuente de identidad.

- Utilice ACLs como mecanismo de seguridad en los documentos ya que agiliza la autorización y ofrece un mejor servicio de búsqueda en general.

Apéndice A

Ejemplo de código cliente de aplicación de confianza en C#

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.IO;
using System.Text;

namespace TrustedApp
{
  class GSAClient
  {

    String GSA_SESSION_ID = "GSA_SESSION_ID";
    String _gsaSessionId = null;
    String _trustedUser;
    String _trustedPwd;
    String _gsaHostName;
    String _endUser;
    String _credentialGroup;
    static void Main(string[] args)
      {
        String gsaHostName = "gsa.acme.com";
        String userName = "trusteduser_a", userPassword = "pwd";
        GSAClient gsaClient = new GSAClient(gsaHostName, userName, userPassword,
          "Default", "enduser_a");
        gsaClient.search("access=a&q=some_keyword&site=default_frontend");
      }

    public GSAClient(String gsaHostName, String trustedUser, String trustedPwd,
      String credentialGroup, String endUser)
      {
        _gsaHostName = gsaHostName;
        _trustedUser = trustedUser;
        _trustedPwd = trustedPwd;
        _credentialGroup = credentialGroup;
        _endUser = endUser;
      }

    String search(String q)
      {
        int iRetry = 0;
        HttpWebRequest request;

        Initiate:
        request = (HttpWebRequest)WebRequest.Create("https://" + _gsaHostName +
          "/search");
        request.Method = "POST";
        request.ContentType = "application/x-www-form-urlencoded";
        ServicePointManager.ServerCertificateValidationCallback = new
        System.Net.Security.RemoteCertificateValidationCallback(AcceptAllCertifications);
        request.Proxy = WebRequest.DefaultWebProxy;
        request.CookieContainer = new CookieContainer();
        if (_gsaSessionId != null)
          {
```

```
    request.CookieContainer.Add(new Cookie(GSA_SESSION_ID, _gsaSessionId)
      { Domain = _gsaHostName });
  }
else
  {
    string authInfo = _trustedUser + ":" + _trustedPwd;
    authInfo = Convert.ToBase64String(Encoding.Default.GetBytes(authInfo));
    request.Headers["Authorization"] = "Basic " + authInfo;
  }
request.Proxy.Credentials = CredentialCache.DefaultCredentials;
((HttpWebRequest)request).KeepAlive = true;

//específico del usuario final
String strRsps = null;
request.Headers["X_GSA_USER"] = _endUser;
request.Headers["X_GSA_CREDENTIAL_GROUP"] = _credentialGroup;

//"X_GSA_USER: usera1" --header "X_GSA_CREDENTIAL_GROUP: Default" -d "access=a&q=
byte[] byteData = UTF8Encoding.UTF8.GetBytes(q);
request.ContentLength = byteData.Length;
try
  {
    using (Stream postStream = request.GetRequestStream())
      {
        postStream.Write(byteData, 0, byteData.Length);
        postStream.Close();
      }

    HttpWebResponse response = (HttpWebResponse)request.GetResponse();
    if (_gsaSessionId == null)
      {
        _gsaSessionId = response.Cookies["GSA_SESSION_ID"].Value;
      }
    StreamReader rsps = new
    StreamReader(request.GetResponse().GetResponseStream());
    strRsps = rsps.ReadToEnd();
    rsps.Close();
    response.Close();
  }
catch (WebException e)
  {
    if (e.Status == WebExceptionStatus.ProtocolError)
      {
        WebResponse resp = e.Response;
          using (StreamReader sr = new StreamReader(resp.GetResponseStream()))
            {
              Console.Write(sr.ReadToEnd());
            }
      }
    if (iRetry == 0)
      {
```

```
                //se presupone que se ha agotado el tiempo de espera de la sesión
                _gsaSessionId = null;
                iRetry++;
                goto Initiate;
              }
            else
              throw e; //si se produce un error, la causa probablemente sea otra.
          }

        return strRsps;
      }

  public static bool AcceptAllCertifications(object sender,
    System.Security.Cryptography.X509Certificates.X509Certificate certification,
    System.Security.Cryptography.X509Certificates.X509Chain chain,
    System.Net.Security.SslPolicyErrors sslPolicyErrors)
      {
        return true;
      }

    }
}
```