GSA | Notes from the Field

Google Search Appliance

Sécurité

Mai 2014

Sécurité

La sécurité constitue une préoccupation majeure dans la conception et la mise en œuvre de solutions de recherche enterprise search intégrant des données issues de différentes sources. Elle fait partie des éléments les plus complexes à gérer dans ce type de projet, en particulier pour la partie Intranet qui requiert généralement les plus hautes exigences en la matière. Il est important de prendre le temps de la réflexion dans ce domaine.

Le présent document décrit en détail les éléments à prendre en compte pour modéliser les exigences en matière de sécurité de façon à créer la meilleure solution possible. Il est essentiel d'appréhender les contraintes du projet dès le départ. En effet, la sécurité est un des domaines les plus difficiles à modifier une fois que la mise en œuvre des autres étapes a débuté.

À propos de ce document

Les recommandations et informations rassemblées dans ce document sont le résultat de notre travail sur le terrain avec de nombreux clients et partenaires dans des environnements variés. Nous tenons à les remercier chaleureusement d'avoir partagé avec nous leurs expériences et leurs observations.

Thèmes abordés	Ce document passe en revue l'ensemble des options de mise en œuvre de GSA (Google Search Appliance) pour vous permettre de mieux comprendre les protocoles de sécurité compatibles avec le produit. Il complète la documentation produit de GSA. Vous y trouverez donc des références aux documents du produit, lesquels détaillent davantage les procédures de configuration des fonctionnalités.
Lecteurs cibles	Ce guide est destiné à toute personne impliquée dans un projet de recherche en entreprise et confrontée à des exigences de sécurité à quelque niveau que ce soit : recensement des exigences, conception du projet ou mise en œuvre de l'éventuelle solution. Ci-dessous figurent les profils impliqués dans le projet qui peuvent tirer parti de ce document : ● Chef de projet ● Chef de projet technique ● Développeur ● Administrateur GSA
Environnement informatique	GSA configuré avec divers mécanismes d'authentification et d'autorisation pour sécuriser les recherches.
Phases de déploiement	Conception de la configuration de la sécurité pour GSA.

Autres ressources	• La documentation produit de GSA fournit des informations exhaustives sur le système de recherche. • Configurer la sécurité du GSA dans le Centre d'aide • Le site Web Learngsa.com fournit des ressources pédagogiques sur le système GSA. • Le Portail d'assistance Google for Work permet d'accéder à l'assistance de Google. • GSA en pratique : Introduction à l'intégration de contenu • GSA en pratique : Manuel des scénarios de déploiement

Sommaire

Chapitre 1 : Conception de la sécurité au sein du GSA

Présentation générale

Les projets de recherche enterprise intègrent des données issues de différentes sources pour permettre aux utilisateurs de trouver facilement les informations qu'ils recherchent. La plupart du temps, notamment dans le cas de projets Intranet, l'accès aux documents des applications source est protégé. Pour fournir aux utilisateurs des résultats pertinents et sécurisés, le moteur de recherche de l'entreprise doit appliquer les mêmes règles d'autorisation que celles des sources où sont stockés les documents.

Le système de recherche rassemble le contenu issu de différentes sources et l'indexe pour faciliter l'accès aux informations. Le système de recherche doit s'appuyer sur les mêmes protocoles de sécurité que ceux utilisés par les applications. Si le projet de recherche de votre entreprise inclut l'indexation du contenu protégé, vous devez, lors de la phase de conception, prendre le temps de modéliser les relations de sécurité entre vos sources de contenu et le système GSA.

Avant la mise en œuvre effective de la sécurité dans le système GSA, prenez le temps de bien comprendre le scénario d'intégration et l'architecture de référence dans leur ensemble. Des règles de sécurité et des protocoles internes sont probablement déjà établis au sein de votre organisation. Vous devez donc trouver les meilleures options pour mettre en œuvre la sécurité dans l'environnement de recherche. En outre, le modèle de sécurité que vous concevez pour le système de recherche doit convenir pour toutes les étapes du projet.

Ce chapitre décrit les processus clés de la recherche sécurisée du système GSA et la manière d'appréhender la conception dans son ensemble.

La recherche sécurisée se compose de trois processus distincts, quoique connexes :

Acquisition du contenu sécurisé	Mécanisme utilisé par GSA pour acquérir la source de contenu sécurisée. Pour obtenir l'accès, GSA doit passer la protection que la source de contenu a mise en place. Bien que rattachée à l'acquisition de contenu, cette étape doit être analysée lors de la conception de la sécurité.
Authentification au moment de la présentation des résultats	Mécanisme utilisé par le système GSA pour identifier les utilisateurs finaux. Il peut d'agir d'un ou plusieurs protocoles d'authentification Internet. C'est le moyen de communication entre GSA et le client (navigateur).
Autorisation au moment de la présentation des résultats	Processus employé par le système GSA pour vérifier si l'utilisateur effectuant une recherche peut accéder aux résultats de recherche.

Acquisition de contenu
Une fois que vous avez modélisé les informations concernant vos sources de contenu, vous pouvez concevoir le(s) mécanisme(s) d'authentification grâce auxquels le système GSA va intégrer chaque source sécurisée. Dans l'étape de conception du projet, il s'agit de la partie du processus qui permet de modéliser l'intégration entre le système de recherche et les systèmes de l'organisation. Le système de recherche autorise l'utilisation simultanée de plusieurs mécanismes d'authentification pour s'adapter à différentes applications lors de l'acquisition des contenus. Le processus implique en principe l'utilisation d'un compte système ou super-utilisateur disposant d'un accès étendu à la source du contenu afin que tous les documents puissent être indexés par le système GSA.

Authentification au moment de la présentation des résultats
L'authentification au moment de la présentation des résultats fait le lien entre le système de recherche et l'utilisateur final. Il peut s'agir du même protocole d'identification que celui utilisé par une des sources de contenu, mais ce n'est pas obligatoire. Plusieurs protocoles d'authentification sont parfois requis afin de permettre l'autorisation de sources de contenu différentes. Vous devez toutefois toujours vous poser les questions suivantes :

- Quels sont les protocoles d'authentification disponibles dans l'environnement du client ?
- Comment puis-je réduire au maximum le nombre de mécanismes d'authentification utilisés lors de la présentation des résultats ? Un seul mécanisme peut-il suffire ?
- Comment puis-je réduire l'impact sur les utilisateurs finaux ? L'authentification peut-elle être silencieuse ?

Autorisation au moment de la présentation des résultats
Chaque source de contenu autorise l'accès aux informations qu'elle contient à l'aide de ses propres règles de sécurité et de sa propre infrastructure. En fonction des informations que vous avez recueillies sur les sources de contenu, sélectionnez les mécanismes d'autorisation en répondant aux questions suivantes :

- Quels sont les mécanismes d'autorisation possibles pour les sources de contenu données ?
- Quel mécanisme offre les meilleures performances ?
- Quels sont les éléments à mettre en œuvre dans le processus d'acquisition de contenu pour que ce mécanisme soit compatible ?

Au moment de la présentation des résultats, l'authentification est réalisée avant l'autorisation. Vous devez toutefois évaluer les options d'autorisation EN PREMIER. Les exigences requises pour l'autorisation déterminent généralement les mécanismes d'authentification qui doivent être pris en compte. Quoi qu'il en soit, ces trois processus sont interconnectés et vous devez étudier les conséquences de chaque décision.

Collecte d'informations

Nous vous recommandons de réaliser les actions suivantes au cours de l'analyse préalable :

- Préciser l'ensemble des exigences relatives à la sécurité, y compris les éventuels besoins ne faisant pour l'instant pas partie du champ d'action du projet, mais susceptibles d'apparaître au cours d'une étape ultérieure.
- Si l'une des exigences concerne l'authentification silencieuse, veuillez vous assurer de sa

faisabilité avant de vous y engager.

- Identifier les mécanismes de sécurité certifiés par votre organisation. Existe-t-il un système d'authentification unique (SSO) ? Le protocole Kerberos est-il activé ?

Modélisez chaque source de contenu en vous servant du tableau ci-dessous. Incluez les informations relatives à la sécurité dans le champ "Mécanismes de sécurité".

Informations sur le système	Nom du système et nom du produit sous-jacent
Description	Description détaillée
Type de contenu	S'agit-il, par exemple, de documents bureautiques, de pages Web ou d'enregistrements de base de données ?
Taille du contenu	Nombre de documents : s'il y a peu de contenus, l'autorisation au moment de la présentation des résultats peut être réalisée par liaison tardive.
Authentification au moment de la présentation des résultats	Le serveur de contenu fait-il appel à l'authentification Windows intégrée ?
	Le serveur de contenu est-il intégré avec un système SSO ?
	Le serveur de contenu dispose-t-il de son propre annuaire utilisateur stocké dans une base de données ou un système LDAP ? Si oui, les noms d'utilisateur sont-ils synchronisés avec l'annuaire de l'entreprise ?
Autorisation au moment de la présentation des résultats	Quelle est la réactivité du serveur de contenu ? S'il n'est pas réactif, la liaison tardive n'est probablement pas envisageable.
	Les autorisations sur les documents sont-elles plutôt permissives ou très restrictives ? Dans ce dernier cas, la liaison tardive est probablement une solution à écarter.
	Existe-t-il une API permettant d'indiquer si un utilisateur a accès à une liste de documents en fournissant le nom de l'utilisateur et les identifiants des documents ? Si tel est le cas, le connecteur ou l'autorisation SAML sont possibles.
	Existe-t-il un moyen de connaître les groupes/rôles/utilisateurs ayant accès à chaque document ? Si la réponse est "oui", l'autorisation LCA est sûrement l'approche à privilégier.

Acquisition de contenu

Les types d'acquisitions les plus courants sont répertoriés ci-dessous. Veuillez noter que le protocole d'authentification utilisé doit être compatible avec la source de contenu. Toutefois, l'acquisition de contenu autorise généralement plusieurs mécanismes d'autorisation.

	Mécanismes d'autorisation possible pour la présentation des résultats	Remarques
Exploration directe	Autorisation SAML, par LCA et par requête Head	Le développement d'un serveur proxy personnalisé peut être requis pour un traitement supplémentaire.
Flux	Autorisation SAML, par LCA et par requête Head	Mise en œuvre simple et individualisée.
Connecteurs	LCA, Connector Authorization	Il peut s'agir d'un connecteur du commerce ou d'un connecteur personnalisé à développer.

Identité simple ou identités multiples

En examinant les sources de contenu, vous devez être capable de répondre à une question cruciale : un seul groupe d'identification (celui par défaut) est-il suffisant ? La réponse à cette question détermine le modèle pour l'authentification au moment de la présentation des résultats. Si chaque utilisateur est associé à plusieurs identités, vous devrez probablement définir plusieurs groupes d'identification. L'existence de différents protocoles d'authentification pour différentes sources de contenu n'implique pas nécessairement plusieurs identités. Par exemple, une source de contenu peut utiliser une authentification par formulaire, alors qu'une autre fait appel à Kerberos. Cependant, si un même annuaire Active Directory est utilisé sur les deux systèmes, il n'existe qu'une seule identité par utilisateur. Le seul cas de figure où GSA peut nécessiter plusieurs identités se présente quand les informations des utilisateurs sont stockées dans des référentiels distincts. À noter toutefois les deux exceptions suivantes :

- Si l'annuaire d'utilisateurs est une copie d'un autre, il n'existe quand même qu'une seule identité par utilisateur. Un seul groupe d'identification est donc suffisant. Par exemple, lorsque Documentum est intégré à Active Directory, une approche consiste à copier tous les utilisateurs dans la base de données Documentum.

- Si les noms d'utilisateur sont exactement les mêmes dans les deux annuaires, un seul groupe d'identification est suffisant, à condition de mettre en place un service de traduction des identités utilisateur dans le cadre du processus d'autorisation.

Sélection d'un mécanisme d'autorisation

L'autorisation et l'authentification au moment de la présentation des résultats sont très étroitement liées. Comme indiqué précédemment, vous devez évaluer les options d'autorisation EN PREMIER, même si l'authentification est réalisée avant l'autorisation au moment de la présentation des résultats. C'est un point crucial qui mérite d'être rappelé. Ce chapitre décrit en détail les connexions qui existent entre ces deux processus.

L'autorisation est toujours étudiée en fonction de la source de contenu. Son objectif est de veiller à ce que les utilisateurs ne puissent afficher que les résultats de recherche qu'ils sont autorisés à voir. C'est le critère le plus important dans la sélection du mécanisme d'autorisation, le second étant les **performances**. Du point de vue des performances, le mécanisme d'autorisation doit satisfaire les exigences suivantes :

- Les résultats de recherche doivent s'afficher aussi vite que possible pour que les utilisateurs finaux bénéficient d'une utilisation optimale. Des études ont montré que, si la recherche prend trop de temps, de nombreuses personnes abandonnent et l'utilisation de l'outil de recherche s'en trouve réduite.
- Les performances doivent être suffisamment élevées pour que des résultats pertinents ne soient pas écartés en raison d'un délai de chargement trop important. Si le délai d'autorisation expire pour certains résultats, ces derniers ont une décision d'autorisation indéterminée et ne s'affichent pas dans la liste des résultats de la recherche.
- Lorsqu'une autorisation de liaison tardive est utilisée, vous devez en minimiser l'impact sur les performances du serveur de contenu.

Pour les projets de déploiement, s'il existe un connecteur fourni par Google ou par un de ses partenaires, l'autorisation est déjà décidée par la conception du connecteur. Vous devez sélectionner un mécanisme d'autorisation uniquement dans les circonstances suivantes :

- Plusieurs connecteurs sont proposés par différentes parties et utilisent des mécanismes d'autorisation différents. De nombreux facteurs entrent en ligne de compte dans le choix du connecteur à adopter, notamment le coût. Le mécanisme d'autorisation n'est qu'un de ces facteurs.
- Un connecteur est parfois compatible avec plusieurs mécanismes d'autorisation. Par exemple, le connecteur Google Search Appliance pour SharePoint est compatible avec trois mécanismes : LCA par URL, Connector et requêtes Head.
- Lorsqu'il n'existe aucun connecteur, vous devez développer un code personnalisé pour intégrer le contenu sécurisé. Cette situation se produit lorsque vous devez prendre en compte toutes les options.

Nous abordons ci-dessous la question de l'autorisation. Les mécanismes sont répertoriés du plus performant au moins performant. GSA traite l'autorisation selon deux approches principales :

- Autorisation de liaison précoce
- Autorisation de liaison tardive

En général, le processus d'autorisation au sein du GSA est plus rapide avec la liaison précoce qu'avec la liaison tardive, mais cela ne veut pas dire que la liaison précoce soit la méthode à utiliser pour toutes les sources de contenu.

Autorisation de liaison précoce

Avec la liaison précoce, l'autorisation est totalement gérée en interne par le système de recherche. La liaison précoce requiert que le système GSA connaisse les règles d'autorisation. Le système GSA s'assure que l'utilisateur a bien l'autorisation d'accéder à un document, sans avoir à contacter un composant de sécurité externe tel que la source de contenu lors de la présentation des résultats.

Le système GSA est compatible avec les deux types de LCA suivants :

LCA par URL

Avec les LCA par URL, chaque document figurant dans l'index peut avoir ses propres règles d'autorisation. L'ajout d'une LCA par URL à un document peut être effectué via des flux, des métadonnées dans le corps HTML ou des en-têtes HTTP personnalisés. Les LCA par URL peuvent inclurent à la fois des utilisateurs
et des groupes. La LCA par URL est généralement la méthode privilégiée, car elle est s'adapte plus facilement au nombre de documents et offre de meilleures performances.

Points à prendre en compte pour l'utilisation des LCA par URL :

- Cette approche s'avère très intéressante si vous avez des règles d'autorisation très pointues et que vous attendez des réponses d'autorisation rapides. Avec les LCA, il est crucial que le processus d'autorisation soit rapide pour des fonctionnalités GSA telles que la navigation dynamique, le filtrage des doublons dans l'annuaire et les clusters de résultats dynamiques.
- Cette approche complique la résolution de l'adhésion aux groupes dans le système de recherche. Cette résolution peut, dans certains cas, être gérée par le système GSA (par exemple, si les groupes figurent dans un annuaire LDAP comme Active Directory). Vous pouvez également créer vos propres processus personnalisés destinés à transmettre les groupes au système de recherche. Dans la version 7.2, une base de données des groupes embarquée est proposée en tant que fonctionnalité bêta et offre une intégration renforcée.
    - Le connecteur Google Search Appliance pour Active Directory Groups est fourni pour résoudre les groupes d'un ou plusieurs domaines Active Directory.
- Il faut également prévoir un délai entre la modification d'un paramètre de sécurité dans la plate-forme source et la notification de cette modification au système de recherche.

- Il est possible de modifier le nombre maximal de principaux pouvant être associés à un document. La valeur par défaut est définie sur 10 000. Le nombre maximum est de 100 000.
    - Le scénario le moins favorable ci-dessous a été testé et le filtrage LCA offre de bonnes performances (délai inférieur à une seconde).
        - 10 000 URL à filtrer
        - Chaque URL comprend 10 000 éléments dans la LCA
        - L'utilisateur effectuant la recherche est membre de 1 000 groupes, mais n'a accès à aucun document. Le système GSA doit donc filtrer toutes les URL qui correspondent au terme recherché.

Règles LCA
Une règle LCA cible la protection de formats d'URL plutôt que d'URL spécifiques. C'est la raison pour laquelle elle peut englober de nombreux documents. Vous pouvez configurer une règle LCA basée sur des formats d'URL en utilisant la console d'administration GSA ou l'API Policy ACL. Utilisez les règles LCA lorsque les règles d'autorisation sont peu nombreuses et que chacune d'entre elles peut regrouper plusieurs URL.

Bien que moins utilisé que les LCA par URL, cet outil très modulable permet de répondre à des situations particulières. Par exemple, s'il existe un groupe défini globalement auquel l'accès à une source de contenu facilement identifiable doit être refusé, la définition d'une règle LCA unique peut s'avérer l'option à choisir. C'est aussi le cas lorsque le système de contenu fait appel à des règles générales d'autorisation. Par exemple, CA SiteMinder permet de définir un contrôle d'accès basé sur des formats d'URL. Ces règles peuvent être facilement transposées en règles LCA.

Dans la version 7.0 de GSA, les règles LCA requièrent la spécification des éléments suivants : domaine, espace de noms, respect de la casse.

Autorisation de liaison tardive

Avec la liaison tardive, le système de recherche ne dispose pas des informations d'autorisation relatives au contenu sécurisé (c'est-à-dire aux LCA). Avant de renvoyer les résultats de recherche à l'utilisateur, le système GSA doit effectuer un contrôle de la sécurité en contactant un composant tiers pour vérifier que l'utilisateur a bien la possibilité de lire le document protégé apparaissant dans les résultats. Le composant tiers répond au système de recherche en renvoyant la décision d'autorisation. Le composant tiers peut être la source de contenu elle-même ou un serveur d'autorisation qui centralise cette décision.

Le système GSA est compatible avec les approches de liaison tardive suivantes :

Connecteurs
Google propose des connecteurs (SharePoint et Documentum, par exemple) qui sont totalement compatibles avec le système de recherche et qui peuvent donc être utilisés dans vos projets pour intégrer des sources tierces. L'environnement de développement Connector Framework, créé par Google, dans lequel ils s'exécutent peut également être utilisé pour créer vos propres connecteurs. Le principal avantage de cette plate-forme en matière de création de connecteurs réside dans le fait qu'elle offre une intégration étroite de la sécurité dans le système de recherche, de la configuration jusqu'à l'indexation.

Connector Framework fournit l'interface SPI permettant à n'importe quel connecteur de mettre en œuvre l'autorisation. L'interface fonctionne par lot (plusieurs documents par appel) afin de fournir les réponses sans avoir à multiplier les aller-retour. D'autres connecteurs développés par des partenaires de Google dans cet environnement de développement utilisent cette approche.

Autorisation SAML

SAML est un environnement de développement basé sur XML et permet de transmettre des informations relatives à l'authentification utilisateur, aux droits et aux attributs. Il s'agit d'un protocole standard qui peut être utilisé pour l'authentification, mais aussi éventuellement, pour l'autorisation. La SPI d'autorisation décrit comment procéder à l'autorisation avec SAML. Vous pouvez utiliser SAML pour l'autorisation sans l'utiliser pour l'authentification et inversement, car ces processus sont tous les deux indépendants. Le système de recherche envoie des requêtes d'autorisation SAML au format XML au service externe que vous avez configuré et le serveur fournit, en réponse, les autorisations pour chaque document.

De nombreux produits d'authentification SAML (IDP) sont commercialisés, mais les fournisseurs de service d'autorisation se font plus rares. SAML Bridge propose cette fonctionnalité en se servant de l'emprunt d'identité Kerberos pour effectuer l'autorisation à l'aide de requêtes Head groupées. Il s'agit d'une ancienne fonctionnalité utilisée à l'époque où le connecteur et l'autorisation LCA n'existaient pas. Le choix s'orientera donc vers cette approche pour un projet personnalisé que vous développez vous-même et non en cas d'utilisation d'un produit existant.

Les autorisations SAML peuvent être gérées par lots pour permettre au système de recherche d'accélérer le processus d'autorisation en envoyant une liste d'URL dans une même requête. Vous pouvez activer cette option dans la console d'administration du GSA, à condition que votre fournisseur d'autorisation SAML soit compatible.

Requêtes Head

Enfin, il est également possible de valider les autorisations en envoyant une requête Head HTTP à la source de contenu. Le système GSA peut envoyer une requête HTTP en se servant de l'URL du document, puis lire la réponse HTTP provenant de la source pour déterminer l'autorisation en fonction des codes d'erreur HTTP suivants :

- 200 : ce code indique que l'utilisateur peut accéder au document. Le système de recherche le considère donc comme une autorisation. Il est également possible de définir des règles d'exclusion dans le système de recherche, car certaines sources de contenu incluent des codes d'erreur HTTP 200 qui comprennent un message d'interdiction d'accès (certaines solutions de portail Web, par exemple).

- Tout autre code d'erreur signifie que l'utilisateur n'a pas accès à ce document spécifique.

Pour vérifier l'autorisation pour tous les résultats, une requête Head est envoyée à chaque document de façon séquentielle jusqu'à ce que le nombre de documents trouvés et autorisés soit suffisant pour remplir au moins une page de résultats de recherche. C'est la raison pour laquelle la requête Head est le mécanisme d'autorisation qui présente les performances les moins bonnes. Elle est généralement utilisée lorsqu'aucune autre solution n'a été trouvée pour extraire la LCA ou vérifier les autorisations à l'aide d'une API.

Connector 4.0(version bêta)

Connectors version 4.0 est un nouvel environnement de développement de connecteurs. Son architecture est complètement différente des versions précédentes. Veuillez noter qu'il ne s'agit pour l'instant que d'une version bêta. Les fonctions de sécurité incluses fonctionnent, elles aussi, différemment des versions précédentes. Voici les principales différences au niveau de la sécurité :

- Un connecteur peut être conçu pour fournir l'authentification et l'autorisation. Le protocole de communication entre le système et le connecteur n'est plus un code XML propriétaire. Le mécanisme d'échange de messages sous-jacent est SAML. Exemple : Google Authentication Adaptor. Ce connecteur fournit l'authentification à l'identifiant Google. Il se configure de la même manière que le fournisseur SAML.

- Un connecteur conçu dans l'environnement de développement 4.0 est compatible avec les LCA par URL.

- Les connecteurs peuvent être conçus pour fournir une résolution de groupe via l'authentification SAML. Toutefois, la résolution de groupe à utiliser en priorité avec GSA 7.2 est la résolution de groupes intégrée (version bêta). La résolution de groupe via SAML fait partie de l'authentification SAML. Ce n'était pas le cas dans le précédent environnement de développement Connector : les connecteurs pouvaient seulement effectuer une résolution de groupe, l'authentification étant effectuée par un autre mécanisme.

Sélection d'un mécanisme d'authentification

En général, lors d'un déploiement, plusieurs mécanismes d'authentification sont à votre disposition. Comme décrit dans le premier chapitre, il est crucial d'utiliser le moins de mécanismes d'authentification possible. Un autre critère est souvent exigé : l'authentification silencieuse. Tous les mécanismes d'authentification ne fonctionnent pas avec tous les mécanismes d'autorisation. Il existe deux types de mécanismes d'autorisation :

- **Ceux pour lesquels l'identifiant utilisateur est requis**
- **Ceux pour lesquels l'identifiant utilisateur n'est pas requis**

Tous les mécanismes d'autorisation requièrent l'identifiant utilisateur, à l'exception des requêtes Head. Le tableau suivant répertorie les mécanismes d'authentification fournissant un identifiant utilisateur :

	Mécanismes d'authentification pour lesquels l'identifiant utilisateur est requis
HTTP Basic/NTLM	Il figure dans la liste sous l'intitulé HTTP Basic/NTLM. Ce protocole d'authentification est utilisé pour vérifier les identifiants utilisateur et s'exécute entre le système GSA et un serveur d'application. Pour l'utilisateur final, il s'agit de l'authentification par formulaire. Une fois que les identifiants ont été validés par l'URL exemple configurée, l'identifiant saisi par l'utilisateur est considéré comme l'identifiant validé.
Certificat client	Le nom distinctif du certificat est transmis en tant qu'identifiant validé.

Kerberos	L'identifiant de l'utilisateur Windows extrait du ticket Kerberos est utilisé en tant qu'identifiant utilisateur validé.
Authentification SAML	L'identifiant validé est transmis par le fournisseur d'identité SAML dans "Subject".
Authentification LDAP	L'identifiant utilisateur validé est celui qui a été validé par le serveur LDAP.
Authentification par formulaire avec déchiffrement de cookie	L'identifiant utilisateur est renvoyé au système GSA par l'outil dé déchiffrement de cookie. L'écriture d'un code spécifique est requise lorsqu'une page Web dynamique simple doit être mise en œuvre pour cette authentification.
Connecteurs	Connector Framework fournit l'interface SPI d'authentification qui renvoie un identifiant utilisateur de confiance. L'authentification doit toutefois être mise en œuvre par le connecteur, ce dernier étant facultatif. Les connecteurs disponibles ne fournissent pas tous l'authentification. Les développeurs du connecteur peuvent même exiger l'utilisation d'un mot de passe. Par exemple, File System Connector 2.x requiert à la fois le nom d'utilisateur et le mot de passe pour exécuter l'autorisation de liaison tardive lorsque des règles de refus s'appliquent aux documents.

Tous les mécanismes d'authentification ci-dessus peuvent être associés aux autorisations SAML, Connector et LCA. Choisissez celui qui est adapté aux exigences du client et qui est le plus simple à mettre en œuvre (veuillez également tenir compte des éventuelles exigences en matière d'authentification silencieuse).

Avec l'autorisation par requête Head, vous ne pouvez pas choisir n'importe quel mécanisme d'authentification, car les requêtes Head sont envoyées par le système GSA à la source de contenu et non par le navigateur du client au système GSA. Les identifiants devant être obtenus par le système GSA au cours du processus d'authentification de l'utilisateur diffèrent en fonction du protocole d'authentification utilisé par la source de contenu.

	Mécanisme d'authentification pour lesquels l'identifiant utilisateur n'est pas requis (requêtes Head)
Cookie	Il s'agit de la situation la plus courante. Le système de recherche transfère les cookies utilisateur pour valider les droits d'accès. L'authentification par formulaire est requise. L'identifiant utilisateur n'étant pas requis, le déchiffrement de cookie n'est pas nécessaire. La règle est configurée sous **Mécanisme d'authentification de connexion universelle > Cookie.**
HTTP	Une règle HTTP Basic/NTLM doit être configurée. La communication entre le navigateur de l'utilisateur final et le système de recherche se fait via l'authentification par formulaire et non par l'intermédiaire du protocole d'authentification HTTP Basic. La règle est configurée sous Mécanisme d'authentification de connexion universelle > HTTP.
Kerberos	Une règle HTTP Basic/NTLM doit être configurée. La communication entre le navigateur de l'utilisateur final et le système de recherche se fait via l'authentification par formulaire. La règle est configurée sous Mécanisme d'authentification de connexion universelle > Kerberos.

Liaison entre l'authentification et l'autorisation

La plupart du temps, vous n'avez besoin de sélectionner qu'un seul mécanisme d'authentification pour valider les utilisateurs. Vous pouvez configurer plusieurs mécanismes d'authentification, mais votre choix doit être justifié et vous devez être conscient de ce qu'il implique. Voici quelques règles à suivre :

- Lorsqu'il existe plusieurs groupes d'identification, vous devez sélectionner un mécanisme pour chacun d'eux. Ce peut être des mécanismes du même type (deux authentifications par formulaire ou deux authentifications SAML, par exemple). Ils peuvent également être différents (une authentification par formulaire et une authentification Kerberos, par exemple).
- Vous pouvez également configurer plusieurs mécanismes d'authentification pour le même groupe d'informations d'identification. Cette pratique est moins courante. La seule exception à cette règle concerne l'utilisation d'un connecteur pour une résolution de groupe. Dans ce cas, un mécanisme d'authentification (silencieux, en général) valide l'utilisateur effectuant la recherche et le connecteur résout l'adhésion aux groupes pour l'autorisation LCA. Les chapitres qui suivent offrent davantage d'informations sur cette situation particulière.
- Tous les mécanismes d'authentification sont déclenchés. Lorsque deux mécanismes d'authentification sont définis pour un même groupe d'informations d'identification, le second est déclenché, même si la première règle a été satisfaite avec un identifiant utilisateur validé.

Pour relier l'authentification à l'autorisation, le système de recherche utilise une fonctionnalité appelée "autorisation flexible", laquelle s'apparente à un tableau de routage des mécanismes d'autorisation. Cette fonctionnalité permet à l'administrateur de configurer le processus d'autorisation pour les documents par format d'URL selon ce qui convient au déploiement. L'autorisation flexible est gérée en configurant des règles d'autorisation. Une règle comprend les éléments suivants : le contenu auquel la règle s'applique (défini par le format d'URL), une identité qui permet de relier la règle à la règle d'identification ou au mécanisme d'authentification, ainsi que d'autres informations spécifiques au mécanisme d'autorisation.

En général, vous n'avez pas à modifier les paramètres d'autorisation flexible. Les paramètres par défaut fonctionnent dans la plupart des cas. Il s'agit d'un tableau de routage qui permet de créer différentes combinaisons entre les règles d'authentification et d'autorisation. En revanche, des principes précis régissent la possibilité d'associer ou non ces dernières :

- LCA par URL
    - Les LCA font partie de l'index et ne peuvent pas être ajoutées ou supprimées à la volée. Le mécanisme LCA par URL ne peut pas être utilisé si les URL ne sont pas associées à des LCA. Le groupe d'identification associé aux LCA est également déterminé au cours de l'indexation, ce qui ne peut pas être modifié dans les paramètres d'autorisation flexible.
    - Lorsque la règle LCA par URL est définie en tant que première règle, l'autorisation s'effectue dans l'index lorsque des résultats concordants sont identifiés. La LCA par URL offre donc de meilleures performances que les autres autorisations. C'est la raison pour laquelle elle apparaît par défaut avant l'autorisation en cache.
    - Si vous définissez un format d'URL spécifique au lieu de "/" ou que vous placez la règle après d'autres règles d'autorisation dans la liste, c'est le Gestionnaire de sécurité qui effectue l'autorisation. Les performances seront donc moindres, car les LCA par URL seront alors évaluées hors de l'index.
- Connecteur
    - Pour qu'un contenu soit autorisé à l'aide de l'autorisation Connector, les URL doivent débuter par "googleconnector://".

Une fois que le système GSA a authentifié un utilisateur à l'aide d'un mécanisme d'authentification configuré, l'autorisation d'accès aux documents est appliquée dans l'ordre de leur définition dans le tableau d'autorisation flexible, en fonction du format d'URL du document. Si plusieurs mécanismes d'autorisation s'appliquent au document, le système GSA parcourt l'ensemble des règles qui s'appliquent dans l'ordre, jusqu'à ce que l'une d'entre elles renvoie l'état "PERMIT" ("Autoriser") ou "DENY" ("Refuser"). Par exemple, si un connecteur envoie des documents accompagnés de LCA, la règle LCA par URL est évaluée en premier. L'état "PERMIT" ou "DENY" renvoyé représente le résultat final. Cependant, si la valeur renvoyée est "INDETERMINATE", la règle "Connector" est utilisée pour évaluer les documents.

Récapitulatif

Dans ce chapitre, nous avons passé en revue le processus de conception de la sécurité pour le projet de recherche de votre entreprise avec le système Google Search Appliance. Ce processus requiert une solide connaissance de la manière dont la sécurité est gérée au sein de votre organisation, ainsi que pour les sources de contenu associées qui font partie du projet. Voici un récapitulatif du processus de conception de la solution :

- Dégagez du temps en amont pour analyser les sources de contenu : définissez la méthode d'acquisition des sources de contenu, le type d'authentification à utiliser, etc.
- Déterminez le nombre de groupes d'identification nécessaires.
- Déterminez le mécanisme d'autorisation à privilégier pour chaque source de contenu.
- Déterminez le jeu de mécanismes d'authentification le plus réduit possible.
    - Si possible, mettez en œuvre l'authentification silencieuse.
    - Si possible, utilisez des composants compatibles et prêts à l'emploi.
    - Ces mécanismes d'authentification seront-ils compatibles avec l'autorisation de la source de contenu appropriée ?
- Configurez les mécanismes d'authentification de connexion universelle.
- Le cas échéant, effectuez les modifications nécessaires dans les règles d'autorisation flexibles.

Chapitre 2 : Utilisation de fonctionnalités prêtes à l'emploi

Dans ce chapitre, nous allons décrire en détail certains mécanismes d'autorisation et d'authentification. Nous passerons également en revue les scénarios les plus courants compatibles avec Google Search Appliance et les produits associés proposés par Google. Nous nous concentrerons sur les scénarios qui ne nécessitent pas d'écrire du code.

Authentification silencieuse

La sécurité informatique a pour but de protéger les applications et les données en fournissant des informations fiables aux utilisateurs, mais de façon sécurisée. En outre, il est important que les mécanismes de contrôle d'accès aient le moins d'impact possible sur les utilisateurs. Par exemple, si un utilisateur a déjà été authentifié par un composant de confiance, les applications doivent se fier à ce processus pour éviter de demander à plusieurs reprises à l'utilisateur de saisir ses identifiants ou de valider son identité. C'est la raison d'être de l'authentification silencieuse : valider l'identité d'un utilisateur dans le système GSA sans lui demander de passer par un processus de connexion supplémentaire.

L'authentification silencieuse peut être mise en œuvre pour un service de recherche comme pour n'importe quelle autre application d'une organisation. Différents mécanismes d'authentification permettent de fournir un service d'authentification silencieuse : des protocoles Kerberos ou NTML, par exemple, ou encore des applications propriétaires telles qu'un système SSO.

Avant de mettre en œuvre l'authentification silencieuse pour votre environnement de recherche, vous devez répondre aux questions suivantes :

- Quelles sont les options d'authentification silencieuse au sein de mon organisation ? Y-a-t-il une option à privilégier ?
- C'est le cas lorsqu'il y a plusieurs options d'authentification silencieuse (par formulaire et via Kerberos, par exemple). Ces options gèrent-elles les différentes identités utilisateur et les différents identifiants nécessaires à l'autorisation ? Vous devez savoir si utiliser l'une d'entre elles suffit ou si vous avez besoin des deux. Déterminez également si l'une peut certifier l'identité de l'autre.
- Existe-t-il plusieurs domaines d'authentification (plusieurs domaines Windows pour Kerberos, par exemple) ? Cette information permet également de modéliser le processus d'autorisation.
- Quelles applications ou sources de contenu devant être intégrés au moteur de recherche utilisent également le mécanisme d'authentification silencieuse ? Vous devriez pouvoir l'utiliser.

Le système de recherche peut être intégré tel quel avec les systèmes/protocoles d'authentification silencieuse suivants :

Authentification par formulaire ou par cookie

L'authentification par formulaire ou par cookie est le processus conduit par un cookie de session, généralement issu d'un système d'authentification unique. Ce type d'authentification est susceptible d'être silencieux si l'utilisateur a déjà été authentifié avant d'accéder à l'application de recherche. Si ce n'est pas le cas, l'utilisateur se voit demander ses identifiants pour créer les bons cookies de session permettant de fournir le service SSO. La section Scénarios d'authentification par cookie de la documentation GSA inclut des détails techniques sur la manière d'intégrer un système SSO. Si la transmission d'un identifiant

utilisateur au système de recherche est également requise, vous devez mettre en œuvre un processus de déchiffrement de cookie.

Kerberos

Kerberos est le protocole utilisé par défaut dans les réseaux Windows. Le système de recherche peut être configuré pour activer Kerberos afin que l'authentification soit transparente pour les utilisateurs.

SAML

De nombreux systèmes SSO sont compatibles avec le protocole SAML et fournissent un processus d'authentification silencieuse. Veuillez noter que le protocole SAML permet à un service externe de certifier au système GSA l'identité de l'utilisateur en toute sécurité. L'authentification entre l'utilisateur et ce service se fait via des protocoles d'authentification standard tels que Kerberos, NTLM ou par cookie. Les cas où l'écriture d'un fournisseur d'identité SAML (IDP SAML) est requise sont rares. Il est beaucoup plus fréquent d'intégrer dans GSA un IDP SAML déjà déployé dans le réseau du client.

Certificats client

Ce scénario est rarement utilisé. Toutefois, dans les environnements où les utilisateurs disposent de certificats client, il est également possible de configurer le système de recherche pour qu'il authentifie les utilisateurs par l'intermédiaire de certificats X.509, lesquels peuvent également offrir une authentification silencieuse aux utilisateurs.

SAML

Le système de recherche est compatible avec l'intégration SAML. Cette norme de sécurité vous permet de créer des processus d'authentification ad hoc hors du moteur de recherche. Si vous concevez un fournisseur d'authentification SAML, vous pouvez coder la logique d'authentification dont vous avez besoin. Si l'utilisateur est correctement authentifié par ce processus externe, l'identité de l'utilisateur est renvoyée au système de recherche.

SAML est une norme de sécurité. Le protocole est donc compatible avec certaines solutions Open Source ou propriétaires. Certains systèmes SSO fournissent une interface SAML. Vérifiez si l'authentification de votre organisation fournit déjà une telle interface d'authentification pour faciliter l'intégration avec le système de recherche. Si tel est le cas, il n'est pas nécessaire de développer ce service.

Sachez qu'il est également possible de configurer un processus d'autorisation SAML en suivant la procédure décrite au chapitre 3,, mais cela n'a aucun lien avec le fait que l'authentification SAML soit configurée ou non.

Vous pouvez vous référer à la documentation produit GSA pour savoir comment configurer SAML dans le système de recherche.

Liaison précoce avec la LCA par URL

Si vous utilisez les LCA pour l'autorisation, notez que, pour que la vérification LCA soit un succès, tous les composants d'une LCA doivent correspondre à l'identité résolue : domaine, principal d'utilisateur, principaux de groupe, espaces de noms des principaux d'utilisateur et de groupe, respect de la casse spécifiée et type de LCA (Permit/Deny).

Résolution de groupe

Contrairement aux autres mécanismes d'autorisation, l'autorisation LCA requiert une étape supplémentaire : la résolution de groupe pour un identifiant utilisateur validé. La résolution de groupe est très importante
pour que la règle LCA de liaison précoce soit compatible avec le système GSA. Un utilisateur peut être membre de plusieurs groupes dans un système de gestion d'identités. C'est la raison pour laquelle une seule et même modélisation d'identité doit être fournie dans le système GSA. Après authentification, le système GSA stocke l'identifiant utilisateur, ainsi que les groupes dont l'utilisateur est membre. Il existe cinq options permettant de résoudre des groupes :

- **Base de données de groupes**(version bêta). À compter de la version 7.2, le système de recherche inclut une base de données interne qui stocke les LCA. Cette fonctionnalité est encore en version bêta. Ses fonctions et son adaptabilité sont limitées. Les adhésions aux groupes doivent être intégrées au système comme le sont les documents dans l'index du système.

- **Connecteurs**. Connector Framework fournit une interface permettant de résoudre les groupes. C'est le développeur du connecteur qui décide de la mise en œuvre de la LCA par URL ou de la résolution de groupe. Parmi les connecteurs compatibles avec Google, SharePoint, Active Directory Groups et Documentum offrent cette fonctionnalité.

- **LDAP**. L'authentification LDAP permet de résoudre des groupes LDAP imbriqués. Son utilisation n'est pas recommandée avec Active Directory (choisissez plutôt le connecteur Active Directory Groups dans ce cas). Vous pouvez en revanche vous en servir avec d'autres serveurs LDAP.

Les trois options ci-dessus peuvent UNIQUEMENT être utilisées pour résoudre des groupes lorsque l'authentification est effectuée par un autre mécanisme. Les deux options ci-dessous permettent de résoudre des groupes dans le cadre du processus d'authentification. Elles ne peuvent pas être utilisées pour la seule résolution de groupe.

- **Déchiffrement de cookie**. Les groupes peuvent être renvoyés dans un en-tête personnalisé, accompagnés de l'identifiant utilisateur. Le déchiffrement doit être intégré au processus d'authentification par cookie.

- **SAML**. Les groupes peuvent être renvoyés dans le cadre du processus d'authentification SAML. Ce dernier doit être intégré au processus d'authentification SAML.

Ces deux mécanismes sont généralement utilisés pour des déploiements qui requièrent un développement personnalisé. Ils sont décrits plus en détail dans le prochain chapitre.

Espace de noms

Le système GSA est compatible avec l'espace de noms LCA. Le concept d'espace de noms a été créé pour éviter les conflits entre les noms d'utilisateurs ou de groupes issus de plusieurs sources dans l'index. Voici un exemple :

L'utilisateur "Jean Dupont" dispose de deux identités et nous avons configuré deux groupes d'identification : jdupont dans GI1 et jeans dans GI2. Dans l'index, toutes les LCA rattachées à Jean Dupont peuvent être associées à l'une ou l'autre des identités. Elles doivent pouvoir être différenciées. C'est la raison pour laquelle l'espace de noms a été créé.

Si le champ d'application principal est l'utilisateur, l'espace de noms est identique au groupe d'identification. Dans les LCA, le principal doit être :

jdupont dans l'espace de noms GI1
ou
jeand dans l'espace de noms GI2

Cependant, si le champ d'application du principal est le groupe, il n'est pas obligatoire que l'espace de noms soit identique au groupe d'identification de l'utilisateur. Tant que l'espace de noms des groupes résolus correspond à ce qui est défini dans la LCA dans l'index, la vérification d'autorisation fonctionne. Voici un exemple :

La première identité de Jean Dupont, jdupont, est celle issue de l'annuaire général Active Directory de l'entreprise. Il y a bien entendu des groupes Active Directory desquels jdupont est membre. Imaginons qu'une des sources de contenu est Plone. Elle est intégrée à Active Directory, mais dispose de ses propres groupes définis. Comment éviter les conflits en cas de groupes ayant le même nom dans Plone et dans Active Directory ? Les groupes d'Active Directory disposent de l'espace de noms GI1. Nous pouvons donner aux groupes de Plone un espace de noms différent ("espace_plone", par exemple). Les LCA figurant dans l'index disposeront des entrées suivantes :

```
<principal namespace="GI1" scope="user" access="permit">jdupont</principal>
…
<principal namespace="GI1" scope="group" access="permit">auteurs</principal>
…
<principal namespace="espace_plone" scope="group"
access="deny">auteurs</principal>
```

Tant que les groupes adéquats peuvent être résolus pour jdupont au cours de la résolution de groupe, après l'authentification, les autorisations appropriées sont appliquées :

GI1:jdupont appartient aux groupes :
GI1:auteurs, espace_plone:auteurs

Analyse de domaine

Les noms de domaine apparaissent fréquemment dans les identifiants utilisateur et les groupes. Le système de recherche dispose d'un champ séparé pour le *domaine* lorsque le principal est stocké dans les cas suivants :

- Une fois l'utilisateur authentifié, l'identifiant validé résolu, ainsi que les groupes associés, contiennent à la fois le nom d'utilisateur et le nom de domaine.
- Le principal figurant dans les LCA du document contient le nom du principal et le nom du domaine, que ce soit pour les utilisateurs ou les groupes.

Les utilisateurs validés peuvent revêtir différents formats en fonction du protocole d'authentification utilisé.

- bob@**google**.com
- **google**\bob

Le système de recherche analyse continuellement ces formats et extrait le nom de domaine et le nom d'utilisateur au cours de l'authentification et de l'indexation LCA. Dans les deux exemples ci-dessus, **google** est extrait en tant que domaine.

La liaison tardive pour les LCA.

Si vous utilisez les LCA pour gérer l'accès aux documents dans le système GSA, vous devez configurer une liaison tardive de secours au cas où les LCA figurant dans l'index ne seraient pas complètement synchronisées à la source de contenu en cas de dépassement de délai. Lorsque la fonctionnalité de liaison tardive de secours est activée pour l'autorisation flexible, le système GSA accepte uniquement la réponse DENY ("Refuser") pour les mécanismes POLICY et LCA par URL. Pour PERMIT et INDETERMINATE, le système GSA applique les autres règles jusqu'à ce que l'une d'entre elles renvoie une décision autre que
la valeur INDETERMINATE. Si aucune ne renvoie cette valeur, le résultat n'est pas présenté à l'utilisateur.

Connecteurs utilisant la LCA par URL

Espace de noms local

Dans Connector Framework, le concept d'"espace de noms local" fait son apparition. Veuillez noter qu'il s'agit d'un connecteur. Pour la définition LCA, il n'existe qu'un seul attribut d'espace de noms. Dans la configuration du connecteur, il existe deux champs d'espace de noms : "Espace de noms global" (similaire au groupe d'identification dans l'authentification) et "Espace de noms local" (nom du connecteur ou d'un autre connecteur configuré pouvant être sélectionné dans la liste déroulante).

Utilisons la source de contenu Plone de l'exemple précédent. Si un connecteur Plone est développé dans Connector Framework avec le nom d'instance "connecteur_plone", voici à quoi les principaux de LCA ressemblent dans les flux envoyés par le connecteur :

```
<principal namespace="GI1" scope="user" access="permit">jdupont</principal>
...
<principal namespace="GI2" scope="user" access="permit">jeand</principal>
...
<principal namespace="GI1" scope="group" access="permit">auteurs</principal>
...
<principal namespace="GI1_connecteur_plone" scope="group"
access="deny">auteurs</principal>
...
```

Le système de recherche concatène les espaces de noms "global" et "local" dans la configuration du connecteur, tout comme l'attribut "espace de noms" figurant dans la LCA envoyée via les flux.

Contournement de l'analyse de domaine

Comme décrit dans la section précédente, le système essaie d'interpréter le format du principal pour en extraire le domaine. Toutefois, une exception existe : lorsque les LCA sont envoyées par les flux, si l'attribut **principal_type** d'un principal a la valeur "unqualified", le domaine n'est pas analysé et le nom est traité comme un littéral, quel que soit son format. Cet attribut et ce comportement représentent une autre option permettant d'éviter les conflits entre les noms des groupes (pour que le connecteur SharePoint reste compatible). SharePoint vous permet de définir des groupes à différents niveaux de la structure hiérarchique d'un site Web. Si nous utilisons la fonctionnalité "espace de noms local" du connecteur, un espace de noms est associé à chaque site. Connector pour Sharepoint de GSA ajoute en préfixe de chaque groupe local SharePoint l'URL du site à laquelle appartient le groupe et affecte à l'attribut "principal_type" la valeur "unqualified". Le système de recherche stocke ces groupes au moment où ils lui sont transmis afin d'éviter tout conflit en cas d'existence d'un nom de groupe identique sur d'autres sites. Voici un exemple de groupes locaux SharePoint envoyés au système GSA dans des flux :

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

En revanche, si un groupe Active Directory est envoyé, ce dernier ressemblera à :

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

Connector 4.0 (version bêta)

Fonctionnement avec les LCA par URL

L'indexation des LCA par Connector 4.0 diffère de celles effectuées dans les versions antérieures :

- Les LCA ne sont pas envoyées dans des flux. Elles sont indexées en tant qu'en-têtes HTTP.
- Si les LCA sont hiérarchiques, elles ne seront pas alignées au même niveau. L'héritage est toujours utilisé.
- L'espace de noms doit être géré par chaque connecteur. Le connecteur système File et le connecteur SharePoint utilisent le nom "*adaptor.namespace*" comme entrée de configuration.
- Le concept d'espace de noms local disparaît : vous êtes libres de spécifier n'importe quel espace de noms. Les LCA de ce connecteur utilisent toutes le même espace de noms. À l'exception du scénario suivant :
    - L'attribut **principal-type** n'est plus utilisé par Connector 4.0. Le champ d'application des groupes SharePoint est ajouté à l'espace de noms et les principaux sont envoyés sans le préfixe. Par exemple, le groupe "Mon groupe SharePoint" figurant dans "http://sharepointhost/sitecollection/" est traité par le connecteur SharePoint comme suit (en partant du principe que le groupe d'identification est "Default") :

      *Espace de noms : Default*_http://sharepointhost/sitecollection/
      Nom du principal : Mon groupe SharePoint

      Si le principal dispose d'un domaine ("mondomaine\mongroupe", par exemple), ce dernier est traité comme suit :

      Espace de nom : Default
      Nom du principal : mongroupe
      Domaine : mondomaine

Authentification

Comme décrit dans le premier chapitre, l'authentification par connecteur utilise le protocole SAML. Connector Framework 4.0 fournit SAML comme trame pour la sécurité. Les connecteurs conçus dans le nouvel environnement de développement doivent fournir leur propre mise en œuvre du processus d'authentification pour la source de contenu ciblée. Dans la console d'administration, procédez comme suit pour effectuer la configuration : sous **Rechercher > Recherche sécurisée > Mécanismes d'authentification de connexion universelle > SAML**, saisissez les valeurs suivantes :

Identifiant du fournisseur d'identité : entrée de configuration **server.samlEntityId** issue du fichier de configuration du connecteur.
URL de connexion : https://connector-host-name:port/samlip
Clé publique : <clé publique du fournisseur d'identité>

Voici quelques remarques sur la mise en œuvre de SAML par le connecteur :

- Plusieurs connecteurs peuvent fournir une authentification. Les identifiants d'entité sont différents.
- Seule la liaison Post est compatible.
- L'extrémité du fournisseur d'identité SAML "samlip" est codée en dur.
- Les groupes peuvent être renvoyés dans l'assertion SAML de l'attribut "member-of".

Autorisation

Dans cette section, la mention "autorisation" fait référence à la liaison tardive en cas d'utilisation de Connector 4.0. Pour configurer l'autorisation, vous devez procéder comme suit : dans la console d'administration, accédez à **Rechercher > Recherche sécurisée > Autorisation flexible**, puis attribuez à **URL du service d'autorisation** la valeur "https://connector-host-name:port/saml-authz".

Sécurité dans un environnement Windows

La plupart des déploiements du système, lequel intègre la recherche de contenu sécurisé, s'effectuent dans un environnement Windows. Google fournit deux produits pour l'intégration : SAML Bridge et Active Directory Groups Connector.

SAML Bridge

Le système de recherche est directement compatible avec l'authentification Kerberos sous Windows, sans nécessiter l'installation d'un composant externe au système GSA. Kerberos est compatible dans tous les environnements Windows. Il est donc recommandé d'utiliser ce mécanisme pour l'authentification silencieuse. Toutefois, il peut s'avérer insuffisant pour les raisons suivantes :

1. Kerberos est énormément dépendant de l'environnement. Par exemple, un appareil client peut ne pas être compatible avec Kerberos. Certains scénarios réseau peuvent aussi ne pas être compatibles. Dans ces situations, les clients Windows natifs se replient sur l'authentification NTLM. Cependant, le système de recherche n'est pas compatible avec cette dernière en natif et elle ne peut donc pas être utilisée.

2. Certaines organisations n'autorisent pas l'utilisation de fichiers de tableaux de clés (keytab) pour Kerberos. GSA utilise un fichier de tableaux de clés pour activer Kerberos.

3. Lorsque le système GSA est activé pour Kerberos et qu'il est utilisé pour l'autorisation par requête Head, il peut uniquement effectuer la délégation non restreinte. Certaines organisations ne l'acceptent pas.

Si vous voulez activer l'authentification silencieuse lorsque Kerberos (ou le fichier de tableaux de clés) ne peut pas être utilisé, vous devez configurer un processus d'authentification externe. Google fournit un outil Open Source appelé SAML Bridge qui fonctionne avec ces scénarios. Il s'agit d'une solution basée sur SAML qui s'exécute dans l'infrastructure Windows. Elle doit donc être installée sur un hôte distinct, capable d'authentifier les utilisateurs avec NTLM ou Kerberos. Pour obtenir des informations détaillées sur la manière de configurer SAML Bridge, consultez le document Enabling Windows Integrated Authentication (Activation de l'authentification intégrée Windows).

Active Directory Groups Connector

Dans un environnement Windows, de nombreuses sources de contenu sont intégrées avec Active Directory. Active Directory utilise les groupes pour contrôler l'accès à certaines ressources. Le connecteur Google Search Appliance pour Active Directory Groups est un outil qui peut être utilisé pour assurer la compatibilité avec la liaison précoce. Il s'agit de l'approche à privilégier par rapport à l'authentification LDAP pour résoudre les groupes nécessaires à la liaison précoce, car l'authentification LDAP ne peut pas être configurée directement sur le système GSA. Bien que l'authentification LDAP puisse également être utilisée pour la résolution de groupes Active Directory, il s'agit d'une "liaison tardive" qui intervient au cours du processus d'authentification. Le système essaie de contacter directement les contrôleurs de domaine pour obtenir les groupes d'un utilisateur. Le connecteur Active Directory Groups peut également effectuer la résolution de groupes en "liaison précoce" : il balaye l'annuaire Active Directory et stocke dans sa propre base de données toutes les informations sur les groupes dont l'utilisateur est membre. Lors de la présentation des résultats de recherche, le connecteur n'a plus qu'à lire cette base de données au lieu de contacter directement les contrôleurs de domaine. Cette méthode offre de bien meilleures performances, notamment dans un environnement vaste comportant plusieurs domaines.

Voici quelques comportements spécifiques et bonnes pratiques en matière de déploiement :

- L'exécution du connecteur dure longtemps (plusieurs jours si l'annuaire Active Directory comporte de nombreux groupes et utilisateurs). Les actions suivantes sont recommandées :
    - Utiliser des instances de connecteurs Groups Active Directory dédiées. Cette action est conseillée, même pour le connecteur SharePoint qui dispose d'une fonctionnalité de connecteur Active–Directory Groups intégrée et peut indexer à la fois le contenu SharePoint et celui des groupes Active Directory.
    - Augmenter la période de balayage. Le balayage s'effectue en dix étapes qui sont reprises dans les journaux. Si les journaux affichent "mise à jour 1/6" et "mise à jour 2/6", mais pas les mises à jour restantes, c'est un signe que le fil de balayage a été interrompu avant de pouvoir se terminer. Vous pouvez augmenter ce délai en modifiant la variable "traversal.time.limit" dans "INSTALLROOT/INSTANCENAME/Tomcat/webapps/connector-manager/WEB-INF/applicationContext.properties".
- Veillez à effectuer la liaison directement vers un hôte de contrôleur de domaine sans équilibrage de charge pour bénéficier des avantages d'un balayage Active Directory par incrémentation.
    - Le connecteur utilise un point de contrôle spécifique au contrôleur de domaine. Pour tirer parti des mises à jour contrôlées, vous devez donc vous connecter au même contrôleur de domaine à chaque requête.
- Utilisez toujours un connecteur externe et une instance de connecteur par gestionnaire de connecteurs.
    - Cette méthode facilite le dépannage et l'application de correctifs.
    - Elle est plus évolutive, car vous pouvez facilement contrôler la consommation des ressources.

- Utilisez une base de données externe pour stocker les informations relatives aux groupes.
    - Cette méthode est plus fiable qu'une base de données embarquée.
    - Une base de données embarquée est étroitement liée à l'instance du gestionnaire de connecteurs. C'est également le seul moyen de résoudre correctement les groupes lorsque plusieurs combinaisons de connecteurs de groupes Active Directory et de connecteurs SharePoint sont utilisés sur plusieurs gestionnaires de connecteurs. Par exemple, lorsqu'il y a plusieurs domaines Active Directory, il doit y avoir un connecteur pour chaque domaine. Pour résoudre les groupes d'utilisateurs issus de domaines distincts ou dans le cas où les adhésions se font sur plusieurs domaines, les informations relatives aux groupes doivent être stockées dans la même base de données et dans les mêmes tables. La configuration de la base de données s'effectue au niveau du gestionnaire de connecteurs. Vous devez faire en sorte que ces derniers utilisent une base de données externe afin que les instances multiples puissent partager les mêmes données associées.

Périmètre de sécurité

Les documents figurant dans l'index du système de recherche peuvent avoir l'intitulé "public" ou "secure" ("sécurisé"). L'intitulé attribué à un document dépend de la façon dont le contenu a été indexé (par exploration ou par actualisation), ainsi que des informations de configuration présentes dans le système GSA. En matière de sécurité, un document indexé fait partie de l'une ou l'autre des catégories suivantes :

Document public	Document sécurisé
● Document exploré en mode public ● Document de flux non sécurisé ● Contenu issu d'une source de contenu sécurisée qui a été marqué comme public à l'aide de la console d'administration GSA	● Document exploré en mode sécurisé ● Document de flux déclaré comme sécurisé

Les utilisateurs peuvent rechercher les documents publics et y accéder sans avoir à s'authentifier. Toutefois, on doit noter l'exception suivante : Dans la version 6.14 du GSA, la fonctionnalité de périmètre de sécurité a été introduite, ce qui garantit que le système de recherche n'affiche pas de résultats lorsque l'utilisateur n'a pas été authentifié. Lorsque le périmètre de sécurité est activé, le système de recherche doit authentifier un utilisateur avec un des mécanismes d'authentification configurés avant d'afficher un quelconque résultat. Si l'authentification échoue, le système GSA n'affiche pas les résultats, même si ces derniers sont publics. Veuillez noter que cette authentification est uniquement effectuée pour les documents marqués comme publics qui n'ont pas besoin d'autorisation.

Pour configurer le périmètre de sécurité, configurez un mécanisme d'authentification. Ce peut être n'importe quel mécanisme décrit dans le chapitre 2. Après quoi, accédez à **Traitement -> Connexion universelle**, puis activez le périmètre de sécurité. Sachez qu'une fois le périmètre de sécurité activé, ce dernier s'applique au système GSA de manière globale et ne peut pas être configuré par collection ou par frontal.

Exemple de recherche sécurisée

Voici les exigences pour l'inclusion dans la recherche de quatre sources de contenu (toutes sécurisées) :

1. SharePoint 2010 avec authentification Kerberos. Un connecteur pour SharePoint compatible avec Google est utilisé pour indexer le contenu.

2. Le contenu Salesforce est intégré avec un fournisseur d'identité SAML utilisant l'authentification par formulaire, mais l'annuaire utilisateur reste Active Directory. Un connecteur Salesforce est déployé pour indexer le contenu avec les LCA. Le connecteur est conçu dans l'environnement Connector Framework de Google. Les documents envoyés commencent par "googleconnector://".

3. Un site Web IIS personnalisé avec authentification Kerberos. Aucune API n'est disponible pour vérifier les autorisations ou accéder aux LCA. GSA explore le contenu directement.

4. Une ancienne application d'entreprise. Les utilisateurs et les autorisations sont stockés dans la base de données. Cette dernière n'est pas intégrée à Active Directory. Il n'y a pas de mappage direct entre les noms d'utilisateurs d'Active Directory et ceux de cette application. Le contenu est indexé par le connecteur de base de données Google. Une déclaration de requête SQL peut être utilisée pour déterminer si un utilisateur a accès ou non aux enregistrements de la base de données dans les résultats de la recherche.

En outre, l'organisation dispose de plusieurs appareils. Certains d'entre eux ne sont pas compatibles avec Kerberos.

Identités utilisateur
SharePoint, Salesforce et le site Web IIS sont associés au même annuaire Active Directory. L'ancienne application, quant à elle, dispose de son propre annuaire. Par conséquent, nous avons donc besoin de deux groupes d'identification : le groupe d'identification "par défaut" pour l'annuaire Active Directory et le groupe d'identification "Ancien" pour l'application d'entreprise.

Autorisation
Lors de la recherche d'une solution, vous devez commencer par l'autorisation. L'utilisation de la LCA par URL s'impose pour le contenu SharePoint et Salesforce. Le connecteur pour base de données de GSA est compatible avec l'autorisation par requête. Nous pouvons donc utiliser l'autorisation par connecteur pour ce type de contenu. Nous devons quand même utiliser la requête Head pour le site Web IIS personnalisé. Ce site utilise Kerberos. Nous pouvons donc utiliser la requête Head avec Kerberos. SharePoint, Salesforce et les anciennes applications nécessitent que l'identité utilisateur soit validée. Le site Web IIS personnalisé ne l'exige pas.

Authentification
Maintenant que nous avons choisi les mécanismes d'autorisation à utiliser, le moment est venu de sélectionner l'authentification pour chaque groupe d'identification. Pour le groupe d'identification "Par défaut", nous ne pouvons pas utiliser Kerberos sur le système GSA, car certains appareils du client ne sont pas compatibles avec ce mécanisme. Nous devons donc nous rabattre sur SAML Bridge. Après une étude approfondie, nous pourrions être en mesure d'utiliser le fournisseur d'identité SAML déjà utilisé par l'intégration Salesforce. Il renvoie la même identité validée et ne nécessite pas l'ajout d'un serveur supplémentaire pour héberger le pont SAML Bridge.

Ensuite, nous devons vérifier si cette stratégie d'authentification suffit ou non à couvrir les exigences en matière d'autorisation du groupe d'identification "Par défaut". Les contenus SharePoint et Salesforce doivent être couverts, car nous obtenons une identité validée qui sera utilisée pour les vérifications LCA. Le site Web IIS personnalisé pose problème si nous utilisons le fournisseur d'identité SAML de Salesforce avec l'authentification par cookie, car aucun ticket Kerberos ne sera disponible pour l'autorisation par requête Head. Pour répondre à cette exigence, nous ne pouvons utiliser SAML Bridge pour l'autorisation que s'il est compatible avec la délégation Kerberos. Il peut effectuer des requêtes Head groupées à l'aide de Kerberos pour un nom d'utilisateur donné. Cependant, cela nécessite le déploiement d'un pont SAML Bridge. Quitte à déployer le pont SAML Bridge, nous pouvez aussi l'utiliser pour l'authentification.

Pour le groupe d'identification "Ancien", nous devons effectuer l'authentification en comparant les identifiants utilisateur stockés dans la base de données. Toutefois, le connecteur GSA pour bases de données ne fournit pas de mécanisme d'authentification. Dans ce cas, la personnalisation est nécessaire pour mettre en œuvre l'interface du gestionnaire d'authentification du gestionnaire de connecteurs dans le connecteur de base de données.

Nous savons désormais quels mécanismes d'authentification nous allons utiliser. Nous pouvons donc configurer les deux règles suivantes dans le mécanisme d'authentification de connexion universelle :

1. **SAML**. Si le groupe d'identification "Par défaut" est utilisé, SAML Bridge doit être configuré en mode de liaison POST. Consultez la documentation Wiki pour obtenir des instructions détaillées.

2. **Connector**. Si le groupe d'identification "Ancien" est utilisé, le connecteur de base de données personnalisé doit être configuré.

Règles d'autorisation flexibles
Pour la plupart des déploiements, nous pouvons laisser telles quelles les trois premières entrées de l'autorisation flexible : PER_URL_ACL, CACHE et POLICY. C'est également le cas pour ce déploiement spécifique. La règle PER-URL-ACL se déclenchera pour le contenu SharePoint et Salesforce, car les LCA sont indexées avec les documents. Nous devons toutefois apporter certaines modifications à la règle CONNECTOR, car la configuration par défaut est uniquement associée au groupe d'identification "Par défaut".

- CONNECTOR
    - Modifiez l'**identifiant d'authentification** et attribuez-lui la valeur “Ancien” (cela équivaut à sélectionner le groupe d'identification).
    - Saisissez le nom du connecteur de base de données dans le champ **Nom du connecteur**.

Nous devons également définir une règle SAML. Bien que SAML Bridge utilise une requête Head pour autoriser les sites Web IIS personnalisés, nous ne pouvons pas compter sur la règle "HEADREQUEST", car c'est au système GSA d'effectuer la requête Head.

- SAML
    - Ce mécanisme doit apparaître juste après la règle CONNECTOR dans le classement de l'autorisation flexible.
    - **L'identifiant d'authentication** doit être "Par défaut" et mapper vers le groupe d'identification.

L'URL du service d'autorisation doit pointer vers le fichier Authz.aspx de SAML Bridge.

Chapitre 3 : L'authentification pour les développeurs

Dans la mesure du possible, vous devez essayer d'utiliser, dans vos déploiements, des produits existants compatibles avec Google ou fournis par les partenaires de Google, ou d'autres produits tiers du commerce. En général, le fait de suivre cette recommandation permet de réduire les risques, ainsi que le coût d'acquisition global. Cependant, certaines exigences peuvent nécessiter de développer des applications
ou des processus personnalisés externes afin de mettre en œuvre l'intégration du contenu et de la sécurité dans le système GSA.

Le système GSA permet d'appliquer les options suivantes pour la conception de processus d'authentification externes personnalisés :

- Authentification par formulaire avec déchiffrement de cookie
- SAML
- Connector Framework
- Trusted Application*Nouveau*

Authentification par formulaire avec déchiffrement de cookie

Si vos systèmes utilisent déjà une authentification par cookie et que vous souhaitez intégrer la sécurité dans le système GSA, vous avez la possibilité de réutiliser et de personnaliser le processus d'authentification pour créer un service d'authentification silencieuse sur le système GSA. Le déchiffrement de cookie est une solution permettant de personnaliser le processus existant d'authentification par formulaire. Il vous permet d'extraire l'identité de l'utilisateur qui se trouve derrière le cookie d'authentification de l'utilisateur et de le transférer au système de recherche.

Le processus de déchiffrement de cookie doit permettre de découvrir qui est l'utilisateur derrière le cookie en contactant une URL externe à l'aide d'API SSO ou de services similaires et d'envoyer les identifiants utilisateur en tant qu'en-têtes HTTP au système de recherche de manière sécurisée. L'utilisateur doit déjà être authentifié par le système SSO qui a créé le cookie de session avant d'atteindre le système de recherche. Dans le cas contraire, l'utilisateur est redirigé vers une page de connexion afin d'établir l'identité avec SSO, après quoi les identifiants sont envoyés au système GSA.

Pour mettre en œuvre le déchiffrement de cookie avec votre système SSO/par formulaire, vous devez configurer une URL externe protégée par le système SSO. Au cours du processus d'authentification, le système GSA contacte cette URL en lui transférant les cookies de session déjà créés par le SSO. Le service externe peut ainsi valider l'identité de l'utilisateur et la renvoyer au système de recherche dans un en-tête HTTP appelé `X-Username` et, le cas échéant, `X-Groups`. Vous pouvez personnaliser ce processus dans son intégralité pour qu'il vous permette de modéliser la sécurité pour le projet de recherche de votre entreprise.

Pour créer le processus de déchiffrement de cookie, vous devez effectuer les actions suivantes :

1. Créez une application Web capable de valider l'identité d'un utilisateur grâce au cookie de session SSO.

Configurez une règle d'authentification par formulaire dans la console d'administration du GSA.

Points clés à prendre en considération
Si vous voulez effectuer une authentification silencieuse avec votre système SSO, prenez note des éléments suivants :

- Un cookie de session ne doit pas être limité à une même adresse IP utilisateur (certains systèmes SSO appliquent cette restriction par mesure de sécurité.

- Le système GSA doit appartenir au même domaine que le cookie de session utilisé par le système SSO. Par exemple, si un cookie utilise le domaine "foo.com", le système GSA doit être configuré au sein de ce domaine ("gsa.foo.com", par exemple). Le navigateur envoie ainsi le cookie SSO au système GSA. Les champs d'application du domaine du GSA et de celui du cookie doivent être déterminés et synchronisés.

- L'application de déchiffrement du cookie doit être une simple application Web écrite dans un language de programmation compatible avec votre serveur Web/d'application (Java ou .NET, par exemple). L'application renvoie le nom d'utilisateur au système de recherche dans un en-tête HTTP personnalisé. Exemple :
  `X-Username: luis.sanchez`

- L'application de déchiffrement de cookie est généralement déployée derrière le système SSO. Concrètement, elle peut être installée derrière le plug-in Web SSO qui authentifie l'utilisateur. Elle transmet généralement cette identité de manière sécurisée à des applications Web de confiance dans un en-tête HTTP.

- Il ne s'agit pas nécessairement d'une application autonome. Il peut s'agir d'une page ASP, JSP ou de toute autre page Web dynamique d'une application existante protégée par le même formulaire de connexion.

- L'"outil de déchiffrement de cookie" ne "déchiffre pas de cookie" à propement parler. Après l'authentification, un cookie est simplement généré et indique que la session utilisateur est valide. Le nom d'utilisateur peut être obtenu de différentes façons : Par exemple, un système SSO propriétaire tel que SiteMinder utilise l'en-tête HTTP_SM_USER pour ajouter l'identifiant utilisateur.

- S'il n'est pas possible d'utiliser un plug-in SSO Web devant une telle application Web personnalisée pour faciliter le déploiement de cette solution, vous pouvez utiliser l'API SSO pour extraire le compte utilisateur qui se trouve derrière le cookie. Ne simulez jamais un système SSO en production
  qui transmet l'identifiant utilisateur dans un cookie en texte brut, car ceci présente un risque élevé de sécurité.

- Sachez que certains systèmes SSO peuvent être configurés pour transmettre ces identifiants dans un en-tête HTTP sans nécessiter le développement d'une application Web.

Résolution de groupe pour la liaison précoce (autorisation LCA)
Des informations relatives au groupe de l'utilisateur en cours d'authentification peuvent être ajoutées à une réponse d'authentification par déchiffrement de cookie. Le processus de déchiffrement de cookie ne change pas. Seuls les en-têtes de la réponse sont modifiés, ces derniers contenant désormais aussi les informations des groupes de l'utilisateur. Tous les groupes résolus avec le mécanisme d'authentification par cookie appartiennent à l'espace de noms global du groupe d'identification sélectionné pour le mécanisme. Si des groupes sont requis pour la session d'authentification, un en-tête HTTP X-Groups est également ajouté :

```
X-Groups: utilisateurs_service, utilisateurs_entreprise
```

SAML

Le système de recherche est compatible avec SAML 2.0, protocole XML pour un fournisseur d'identité externe. Dans certains cas, vous devez développer un fournisseur d'identité SAML personnalisé. Sachez que la conception de ce dernier en totalité est chronophage. Il est préférable de partir d'un code existant tel qu'OpenSAML.Google fournit également le projet Open Source SAML Bridge pour l'authentification silencieuse avec les technologies Windows.

Gardez à l'esprit qu'il existe deux manières différentes de configurer l'authentification SAML avec le système de recherche :

- Liaison d'artefacts HTTP : le navigateur est utilisé en tant que principal mécanisme de communication entre le système GSA (fournisseur de service) et le fournisseur d'identité (votre serveur SAML).
- Liaison POST HTTP : requiert un mécanisme de confiance entre le système GSA (fournisseur de service) et le fournisseur d'identité (votre serveur SAML).

Ce document n'a pas pour vocation de vous guider dans l'exécution de cette configuration, car cette étape est déjà abordée dans la documentation GSA. Le tableau suivant contient néanmoins des conseils sur le choix de la liaison SAML à utiliser.

	Liaison d'artefacts HTTP SAML	Liaison POST HTTP SAML
Exigences requises	Plusieurs redirections transparentes sont requises. Une relation de confiance est requise entre le navigateur et le fournisseur d'identité SAML/GSA.	Vous devez créer un lien de confiance entre le système GSA et le fournisseur de service, en plus de celui existant entre le navigateur et les autres serveurs.
Configuration	Des connexions HTTP sécurisées entre le client possédant le système GSA et le fournisseur d'identité sont souhaitables. Une URL supplémentaire doit être configurée sur le système GSA (URL de résolution des artefacts) afin de fournir les informations d'authentification finales au système GSA.	Outre les connexions SSL entre le client et les serveurs, vous devez également configurer des certificats entre le système GSA et le fournisseur d'identité. Ces derniers permettent au système GSA d'obtenir les informations d'authentification directement aurpès du fournisseur de service.
Infrastructure de clé publique	Cette méthode est moins complexe du point de vue de la sécurité, car elle ne requiert pas nécessairement une solution d'infrastructure de clé publique.	En revanche, des certificats de confiance doivent être émis. L'infrastructure de clé publique est donc nécessaire. Sachez que le même résultat peut être obtenu en utilisant une solution OpenSSL.
Haute disponibilité	Il est plus difficile de fournir une solution hautement disponible pour le système GSA et le fournisseur d'identité, en raison du grand nombre de redirections du navigateur et du maintien requis de l'artefact entre plusieurs appels.	Une disponibilité élevée peut facilement être mise en œuvre.
Fournisseur d'identité SAML	Si vous devez développer complètement un fournisseur d'identité SAML basé sur la liaison d'artefacts, cela peut être plus complexe, car un service supplémentaire est alors requis (URL de résolution des artefacts). Vous trouverez sur Internet des environnements de développement Open Source comme OpenSAML, ainsi que de nombreux exemples de code.	Le développement complet d'un fournisseur d'identité SAML peut s'avérer plus simple, mais il requiert d'intégrer la gestion des signatures numériques dans votre code.

En général, il est préférable d'utiliser une liaison POST HTTP SAML, car elle fournit une solution plus fiable et plus simple, surtout en termes de disponibilité élevée.

Résolution de groupe pour la liaison précoce (autorisation LCA)
La réponse d'authentification SAML peut contenir d'autres éléments permettant de renvoyer au système GSA les groupes de l'utilisateur authentifié. Tous les groupes résolus avec le mécanisme d'authentification SAML appartiennent à l'espace de noms global du groupe d'identification sélectionné pour le mécanisme.

Exemple de réponse SAML :
Dans cet exemple, vous pouvez voir que la réponse SAML contient à la fois le nom d'utilisateur ("Subject") et une déclaration d'attribut "member-of" contenant les groupes résolus de l'utilisateur.

```
<Assertion Version="2.0"
        ID="blabla2"
        IssueInstant="2011-01-01T14:38:05Z"
        xmlns="urn:oasis:names:tc:SAML:2.0:assertion">
  <Issuer>ac.corp.entreprise.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>employés-fr</AttributeValue>
      <AttributeValue>bureau-Paris</AttributeValue>
    </Attribute>
  </AttributeStatement>
</Assertion>
```

Choisir entre le déchiffrement de Cookie et SAML

Si vous devez personnaliser votre processus d'authentification, il est important de différencier le déchiffrement de cookie et SAML, afin de pouvoir planifier la meilleure approche avant de commencer le projet.

	SAML	Déchiffrement de cookie
Intégration	Certains systèmes d'authentification unique offrent une interface d'authentification SAML pouvant être intégrée au système.	Certains systèmes d'authentification unique peuvent être facilement intégrés par déchiffrement de cookie.
Complexité	Le développement d'un fournisseur SAML dans sa totalité peut s'avérer complexe.	Le développement pour le système d'une solution de déchiffrement de cookie est plus abordable.
Authentification	Le navigateur (utilisateur) et le fournisseur de service interagissent. Ce dernier peut donc être utilisé avec n'importe quel protocole d'authentification point à point (Kerberos ou NTLM, par exemple).	Au cours du processus d'authentification, le système contacte l'URL modèle sans que l'utilisateur n'ait à intervenir. Le processus n'est donc valide qu'avec une approche d'authentification par cookie.

Pour obtenir les informations techniques sur la façon de mettre en œuvre ces deux approches, consultez les articles Déchiffrement de cookie et SPI d'authentification et d'autorisation. Il est important de comprendre l'interaction entre ces deux processus afin de les mettre en œuvre correctement.

Connector Framework pour la résolution de groupe

Connector Framework fournit également une interface pour l'authentification utilisateur. Toutefois, il ne s'agit pas d'un mécanisme d'authentification silencieuse. L'authentification par connecteur n'est donc pas recommandée. Par contre, le connecteur peut servir à fournir une résolution de groupe pour la liaison précoce, ce qui s'avère beaucoup plus utile. Un mécanisme d'authentification silencieuse (Kerberos, SAML ou outil de déchiffrement de cookie, par exemple) est couramment associé à une résolution de groupe par connecteur.

Compatibilité de l'interface
Connector Framework définit l'interface suivante à mettre en œuvre par un développeur de connecteur :

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

Lorsqu'un connecteur est configuré avec des mécanismes d'authentification de connexion universelle, l'option "Effectuer uniquement une résolution de groupe" est disponible.

Lorsque le connecteur est prévu pour fournir à la fois l'authentification et la résolution de groupe, la mise en œuvre peut ne pas prendre en compte les éléments transmis par le système GSA dans l'objet AuthenticationIdentity et renvoyer un nom d'utilisateur validé accompagné de groupes au système GSA par l'intermédiaire de l'objet AuthenticationResponse. Voici le constructeur de l'objet AuthenticationResponse :

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

La variable "valid" indique si l'authentification a fonctionné ou non dans le cas où la méthode "authenticate()" est employée. La variable "data" est conservée en vue d'une future utilisation.
La collection "groups" est utilisée pour conserver les groupes utilisateur de la classe suivante :

```
public class Principal;
```

Cette classe stocke les informations concernant les groupes : nom, espace de noms, respect de la casse et type de principal. Ces informations sont les attributs d'une règle LCA par URL. Le dernier attribut "principle_type" est utilisé par Connector Framework pour introduire un concept d'"espace de nom local". Consultez la section Présentation de l'espace de noms pour obtenir des informations sur la manière dont les espaces de noms doivent être mappés. Voici l'utilisation qui en est faite au cours du balayage :

1. Le connecteur est configuré dans la console d'administration du GSA avec deux espaces de noms : global (groupe d'identification) et local.
2. Le connecteur rassemble tous les groupes et crée l'objet **Principal**. Si tous les groupes appartiennent à une seule source de contenu, la valeur "unqualified" est affectée à l'attribut "principal_type". Le nom de l'espace de noms local est ajouté en préfixe du nom du groupe, comme c'est le cas avec un domaine.
3. Le gestionnaire de connecteurs transpose les objets **Principal** dans la définition LCA des flux XML en transférant les propriétés des objets **Principal** dans les attributs LCA correspondants.

Compatibilité de la base de données
Il existe deux options de conception permettant aux connecteurs de résoudre des groupes au cours de la présentation des données :

1. Si une API est disponible, un connecteur peut envoyer une requête à l'application dans laquelle les adhésions aux groupes sont stockées. Le connecteur compatible avec Google pour Documentum (version 3.2) est compatible avec la liaison précoce et fait appel à cette approche.
2. Les utilisateurs, les groupes et leurs relations peuvent être découverts en amont et enregistrés dans l'espace de stockage propre au connecteur. Au cours de la présentation des résultats, le connecteur lit dans cet espace de stockage et fournit le résultat au système. Le connecteur Active Directory Groups fourni par Google et le connecteur SharePoint en sont des exemples. Il arrive que ce soit la seule option disponible. Par exemple, lorsqu'il existe plusieurs domaines Active Directory, il peut s'avérer très long d'envoyer une requête à partir de chacun d'eux au cours de la présentation des résultats. Un appel en temps réel n'est donc pas possible.

Connector Framework est compatible avec la base de données. Le fichier de configuration "applicationContext.properties" dispose de paramètres de configuration JDBC pour différentes bases de données. Connector Framework intègre une base de données H2 intégrée. Lorsque vous développez des connecteurs, vous pouvez stocker les adhésions utilisateur dans la base de données à l'aide de la seconde approche décrite ci-dessus.

Trusted Application(version bêta)

Trusted Application est très souvent employée pour fournir un service de recherche lorsque le système GSA se trouve derrière un portail. L'interface utilisateur de recherche est intégrée au portail et les utilisateurs n'interagissent pas directement avec le système. Dans ce cas, la difficulté consiste à transmettre les identifiants utilisateur au système GSA sans que l'utilisateur ait à se connecter. Pour y parvenir, la page de recherche doit simuler le comportement du navigateur lors de l'interaction avec le système afin d'obtenir les identifiants de l'utilisateur depuis la session existante du portail et de les transmettre au système. Les applications de portail utilisent différents protocoles d'authentification et les méthodes permettant de gérer les identifiants des utilisateurs finaux sont variées. Ce processus peut donc être difficile à mettre en œuvre.

Afin de faciliter l'intégration pour les développeurs de portails, le système Google Search Appliance a ajouté une nouvelle fonctionnalité dans la version 7.2 : Trusted Application. Le concept est simple : les portails utilisent un compte utilisateur de confiance préconfiguré pour établir la session avec le système GSA en utilisant des protocoles d'authentification simples, mais limités. Ce compte utilisateur de confiance permet également d'envoyer des requêtes de recherche sécurisées à l'aide d'un nom d'utilisateur final. Les éléments suivants permettent de contourner la difficulté rencontrée plus haut :

1. Les protocoles d'authentification simples sont faciles à gérer. L'authentification Basic et le déchiffrement de cookie sont les deux mécanismes compatibles.
2. Seuls les noms d'utilisateur sont requis. Cela permet donc de résoudre le problème qui surgit lorsque des mots de passe utilisateur ne sont pas disponibles dans une session utilisateur établie dans le portail.

L'identité des utilisateurs finaux est envoyée à l'aide des deux en-têtes HTTP personnalisés suivants :

X_GSA_USER : stocke le nom d'utilisateur**.**
X_GSA_CREDENTIAL_GROUP : permet de déterminer le groupe d'identification de l'utilisateur final.

Points clés à prendre en considération

1. Lorsqu'une recherche sécurisée est effectuée, tous les mécanismes d'authentification configurés sont déclenchés, y compris celui qui a été configuré pour Trusted Application.
2. Vous pouvez utiliser le langage de programmation de votre choix : Python, Java ou C#.
3. L' "utilisateur de confiance" doit être validé. Le système GSA est compatible avec le déchiffrement de cookie et l'authentification Basic. Par conséquent, le serveur de contenu du système principal affecte les performances. Vous devez, autant que faire se peut, éviter d'effectuer l'authentification.
4. Vous pouvez utiliser le cookie de session GSA renvoyé suite à un appel au système GSA pour de futurs appels.
5. Chaque requête de recherche doit être accompagnée par le nom de l'utilisateur final et le groupe d'identification afin d'obtenir les résultats associés au bon utilisateur.

6. La résolution de groupe est déclenchée au cours du processus d'authentification de l'utilisateur final. Si aucune résolution de groupe n'est configurée, elle est invoquée. Comme la résolution de groupe a également des répercussions sur les performances, vous devez éviter d'effectuer une résolution de groupe à chaque appel de l'API.

7. Lorsque le même utilisateur final effectue une autre recherche à l'aide du même cookie de session, si ce dernier est encore valide, la résolution de groupe ne sera pas déclenchée une nouvelle fois. Lorsqu'un autre utilisateur final effectue une recherche pour la première fois, une résolution de groupe est déclenchée pour cet utilisateur, même si le cookie de session GSA est identique.

8. Lorsque la session de l'utilisateur de confiance expire (le cookie a expiré en fonction du paramètre "Délai avant expiration de la session" sous **Recherche sécurisée -> Contrôle d'accès**), GSA renvoie l'erreur suivante : "Le serveur distant a renvoyé une erreur : (502) Passerelle erronée."

9. Lorsque la session de l'utilisateur de confiance est valide (le délai avant expiration de la session n'a pas été dépassé), mais que la durée de confiance du mécanisme d'authentification expire, le système effectue une autre authentification à l'aide des identifiants de l'utilisateur de confiance. L'appel prend alors autant de temps que le premier appel ayant renvoyé le cookie de session GSA actuel.

Bonnes pratiques

1. Si l'intégration s'effectue avec un portail, le cookie de session GSA renvoyé doit être stocké dans la session active de l'utilisateur du portail. Concrètement, les utilisateurs du portail auront tous une session GSA différente et toutes doivent être stockées. Ces cookies de session GSA seront réutilisés pour le même utilisateur final afin d'éviter l'exploitation inutile des ressources pour l'authentification de l'utilisateur de confiance.

2. Le meilleur scénario en termes de performances se présente lorsque la session de GSA reste valide pendant toute la durée de la session de l'utilisateur du portail. Toutefois, cela n'est pas garanti : l'utilisateur peut passer du temps à parcourir le portail avant d'effectuer une autre recherche. Votre code doit inclure la gestion du cas où un nouvel appel est effectué alors que le cookie de session de GSA a expiré.

3. Faites en sorte que la durée Trust du mécanisme d'authentification soit identique au délai avant expiration de la session. Par défaut, le délai avant expiration de la session s'élève à 1 800 secondes. Vous évitez ainsi que les performances soient affectées par une autre authentification implicite utilisant l'identifiant de l'utilisateur de confiance.

4. Pour être transmis, le nom de domaine doit être ajouté au nom de l'utilisateur final. Dans le cas contraire, l'appel échoue.

Veuillez vous référer à l'annexe A pour voir un exemple de client développé en C#. Une instance de la classe doit être créée pour chaque session de l'utilisateur du portail. Un second essai doit être effectué lorsque la session de GSA expire.

Authentification avec Connector 4.0 **(version bêta)**

Un connecteur doit uniquement fournir la mise en œuvre pour l'interface suivante :

public interface AuthnAuthority

et enregistrer cette dernière avec :

AdaptorContext.setAuthnAuthority()

Pour vous référer à une mise en œuvre existante, vous pouvez consulter Google Authentication Adaptor.Après l'authentification, un objet appartenant à la classe **AuthnIdentity** est renvoyé. Il contient le nom d'utilisateur et, le cas échéant, les groupes ou le mot de passe.

Chapitre 4 : L'autorisation pour les développeurs

Présentation générale

Un moteur de recherche d'entreprise doit renvoyer des résultats fiables à l'utilisateur, mais uniquement ceux auxquels ce dernier a accès. Pour y parvenir, le processus d'autorisation s'applique à chaque document sécurisé figurant dans l'index. Ce chapitre traite des solutions personnalisées disponibles pour le développement du processus d'autorisation dans le projet de recherche de votre entreprise avec Google.

La section Sélectionner une approche d'autorisation a présenté les options principales suivantes pour la conception d'un processus d'autorisation personnalisé :

- LCA par URL
- Règles LCA
- Autorisation SAML
- Connecteurs

Les sections suivantes fournissent davantage d'informations sur l'utilisation de ces options dans une solution personnalisée.

LCA par URL

Lors de l'utilisation de la liaison précoce dans un connecteur ou un flux personnalisé, la plus grande difficulté réside dans la simulation du modèle d'autorisation du système cible. Le modèle de sécurité de chaque système est différent.

Les règles LCA peuvent être associées aux documents de deux manières : en tant que métadonnées dans des en-têtes HTML ou par l'intermédiaire d'en-tête HTTP personnalisés. Toutefois, seuls les flux vous permettent de spécifier l'ensemble des attributs LCA possibles. Google Connector Framework se base sur les flux. Cette section couvre donc le cas où les LCA sont envoyées par un connecteur. Consultez la section Spécification des LCA par URL pour obtenir des informations sur la manière de définir complètement la règle LCA. L'héritage LCA compte parmi les fonctionnalités les plus importantes offertes par GSA pour simuler différents modèles de sécurité.

Grâce à l'héritage LCA, les modifications de LCA sont traitées plus efficacement. Il n'est plus utile d'étendre les LCA et de les relier à chaque niveau dans une hiérarchie. Les modifications de LCA sont donc traitées de manière plus efficace, car vous n'avez qu'à réindexer le niveau sur lequel l'autorisation a changé.

L'attribut "inheritance-type" permet de modéliser les différents mécanismes de sécurité de chaque système de contenu. Dans une chaîne d'héritage, la vérification d'autorisation dans la liste est toujours réalisée de la fin vers le début et les autorisations sont évaluées en fonction du type d'héritage qui a été défini :

- PARENT_OVERRIDES
    - L'autorisation de la LCA parent est prioritaire sur la LCA enfant, sauf si l'autorisation parent est INDETERMINATE (indéterminée). Dans ce cas, l'autorisation enfant prend le dessus. Si les autorisations parent et enfant sont toutes deux INDETERMINATE, l'autorisation est INDETERMINATE (indéterminée).
- CHILD_OVERRIDES
    - L'autorisation de la LCA enfant est prioritaire sur la LCA parent, sauf si l'autorisation enfant est INDETERMINATE (indéterminée). Dans ce cas, l'autorisation parent prend le dessus. Si les autorisations parent et enfant sont toutes deux INDETERMINATE, l'autorisation est INDETERMINATE (indéterminée).
- AND_BOTH_PERMIT
    - L'autorisation est PERMIT (accepter) uniquement si les autorisations LCA parent et enfant sont toutes deux PERMIT. Dans le cas contraire, l'autorisation est DENY (refuser).

Exemple de chaîne d'héritage
URL

- "FileUrl" (USER:jean access:PERMIT type:LEAF) hérite de
- "FolderUrl" (GROUP:fr access:PERMIT type:CHILD_OVERRIDES) hérite de
- "ShareUrl" (GROUP:interne access:DENY type:PARENT_OVERRIDES

Décisions d'autorisation

- Identité PERMIT (USER:jean, GROUP:fr)
    - PERMIT par ACL URLFichier, non remplacée = PERMIT
- Identité PERMIT (USER:dupont, GROUP:fr)
    - INDETERMINATE + PERMIT + non remplacé = PERMIT
- DENY (USER:adam, GROUP:fr, GROUP:internes)
    - INDETERMINATE + PERMIT + DENY (remplacé) = DENY

Les LCA peuvent être "Free" ("libres") ou "Bound" ("limitées"). Les LCA associées à des documents indexés sont "Bound" (limitées). Les LCA "Free" (libres) représentent les éléments autres que les documents. Par exemple, certains systèmes de contenu définissent des objets d'autorisation pouvant être utilisés par différents documents. Les LCA sont conservées dans ces objets spéciaux et non dans les documents. Les systèmes de contenu tels que les systèmes File, disposent de hiérarchies et les LCA peuvent être définies sur des dossiers autres que des documents. Les LCA "Free" (libres) peuvent être utilisées dans les deux scénarios. Elles ne sont pas comptabilisées en tant que documents indexés et ne font donc pas l'objet d'une licence GSA.

Exemple de LCA "Free" ("libre")

```
<group>
  <acl url='http://hôteexemple.corp.google.com/'
      inheritance-type="child-overrides" inherit-
from='http://corp.google.com/'>
    <principal scope="user" access="permit">edouard</principal>
    <principal scope="user" access="deny"> vincent</principal>
    <principal scope="user" access="deny"> ben </principal>
    <principal scope="group" access="permit">nobles</principal>
    <principal scope="group" access="deny">dramaturges</principal>
  </acl>
  ...
  ...
</group>
```

Dans cet exemple, `http://hôteexemple.corp.google.com/` est une LCA libre qui hérite de `http://corp.google.com/` et définit d'autres principaux. Le type d'héritage de la LCA est "child-overrides". Cette LCA est donc remplacée par son enfant, le cas échéant.

Autorisation SAML

Vous pouvez totalement personnaliser le processus d'autorisation à l'aide d'un fournisseur SAML qui résout l'autorisation. Il est préférable de concevoir un tel processus d'autorisation SAML avec le langage de programmation que vous maîtrisez le mieux. La requête d'autorisation SAML est une requête au format XML envoyée par le système de recherche à l'URL du service que vous avez configuré dans la console d'administration. Cette requête contient des informations sur l'utilisateur et les URL à autoriser. SAML est également compatible avec les processus groupés, afin que plusieurs URL puissent être envoyées en même temps, ce qui est particulièrement souhaitable si on utilise cette approche pour accroître les performances en évitant les interactions pour l'autorisation.

Le Guide SPI Enterprise pour l'autorisation et l'authentification contient de plus amples informations sur le format XML SAML. Vous pouvez utiliser ce dernier pour concevoir un processus d'autorisation SAML personnalisé. Vous devez mettre en œuvre le service s'exécutant sur le serveur d'application externe. Ce dernier analyse la réponse, extrait les informations sur la détention par l'utilisateur des droits d'accès au document et renvoie une réponse au format XML au système de recherche. Par exemple, le pont SAML Bridge permet d'effectuer l'autorisation groupée du contenu traité avec Kerberos à l'aide des requêtes Head.

Points clés à prendre en considération
Éléments à prendre en compte pour l'utilisation de l'autorisation SAML :

- Le principal avantage de la mise en œuvre de ce modèle d'autorisation est que vous pouvez contrôler le processus de sécurité dans son ensemble au moment de la présentation des résultats.
- Le principal inconvénient de cette approche repose dans sa relation intrinsèque avec la méthode de liaison tardive. Ceci étant, la gestion de l'autorisation peut prendre plus de temps. En revanche, le traitement en groupe permet de réduire ce délai de traitement.

Autorisation avec Connector Framework

Une autre option pour modéliser la sécurité consiste à mettre en œuvre un connecteur personnalisé. Comme décrit dans ce document et dans la documentation GSA, un connecteur peut être créé pour "balayer" un contenu public ou sécurisé, ou transmettre ce dernier par l'intermédiaire d'un flux au système de recherche. Il assure également la compatibilité avec l'autorisation et l'authentification au moment de la présentation des résultats. Nous avons passé en revue l'utilisation des connecteurs à l'aide de la LCA par URL. À présent, nous étudions l'utilisation des connecteurs dans le cadre d'une autorisation en tant que mécanisme de liaison tardive.

Compatibilité de l'interface
Connector Framework définit l'interface suivante à mettre en œuvre par un développeur de connecteur :

```
public interface AuthorizationManager;
```

Il dispose de la méthode d'autorisation suivante :

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

"docids" est une collection d'identifiants de documents uniques destinés à être comparés pour les résultats de recherche. Plusieurs identifiants de documents sont transmis par le système au connecteur. Lorsque suffisamment de documents sont autorisés, le système cesse d'appeler le connecteur. Dans le cas contraire, le système continue d'appeler l'API (à chaque appel, le nombre d'identifiants de documents est supérieur à celui de l'appel précédent) jusqu'à ce que le temps alloué expire, jusqu'à ce qu'il n'y ait plus d'identifiants de documents ou jusqu'à ce que suffisamment de documents aient été renvoyés avec l'attribut "PERMIT" à l'utilisateur effectuant la recherche.

AuthenticationIdentity comprend l'identité validée de l'utilisateur. Selon le protocole d'authentification utilisé, cet attribut peut inclure le nom d'utilisateur, le domaine, ou même le mot de passe (si le protocole d'authentification déployé récupère le mot de passe). Les informations minimales à inclure dans **AuthenticationIdentity** doivent être décidées lors de la mise en œuvre d'un connecteur.

Autorisation Connector 4.0 **(version bêta)**

Un connecteur doit uniquement fournir la mise en œuvre pour l'interface suivante :

```
public interface AuthzAuthority
```

et enregistrer cette dernière avec :

AdaptorContext.setAuthzAuthority()

Proxy Web

Les options décrites ci-dessus sont les plates-formes les plus couramment utilisées pour mettre en œuvre la partie "sécurité" de l'interconnexion avec une source de contenu. Il en existe d'autres, telles que l'utilisation d'un proxy Web pour gérer l'autorisation.

Dans ce cas, l'autorisation est centralisée dans un proxy Web. Pour le balayer, toutes les URL doivent être réécrites. Le système de recherche envoie donc des requêtes Head HTTP pour valider la sécurité avant d'afficher les résultats.

Points clés à prendre en considération
L'utilisation d'un proxy Web s'apparente à celle d'un fournisseur d'autorisation SAML, mais elle présente les inconvénients suivants :

- Les requêtes d'autorisation ne sont pas groupées.
- Les URL doivent être réécrites pour passer le proxy Web pour l'autorisation. Exemple : `http://proxy.corp.com/proxy?returnPath=http//cont.corp.com/doc.html`
- Ces URL ont été réécrites et sont stockées dans l'index. Elles doivent ensuite retrouver leur libellé d'origine dans le frontal de recherche.

Récapitulatif

Dans ce chapitre, nous avons passé en revue le processus de conception de la sécurité pour le projet de recherche de votre entreprise avec Google Search Appliance. Ce processus requiert une solide connaissance de la manière dont la sécurité est gérée au sein de votre organisation, ainsi que pour les sources de contenu associées qui font partie du projet. Vous devez consacrer suffisamment de temps à l'analyse de ce scénario et à la modélisation de l'autorisation et de l'authentification dans le système de recherche.

Présentation générale des bonnes pratiques relatives à la sécurité

- Analysez consciencieusement les éléments suivants en amont :
    - Choix des fournisseurs d'identité à intégrer pour l'authentification
    - Méthode d'autorisation des documents pour chaque source de contenu intégrée au système GSA

- Lorsque cela est possible, utilisez des composants du commerce compatibles pour intégrer la sécurité sur GSA, par exemple :
    - Kerberos
    - Google Search Appliance SAML Bridge pour Windows
    - LDAP
    - Google Search Appliance Connector pour Active Directory

- Modélisez chaque fournisseur d'identité que vous devez intégrer avec un groupe d'identification.

- Classez les groupes d'identification par système de sécurité d'entreprise (fournisseurs d'identité) et associez-les aux sources de contenu.
    - Si possible, utilisez un seul groupe d'identification par fournisseur d'identité.
    - Les groupes d'identification doivent être mappés à des mécanismes d'identité uniques, pas nécessairement des sources de contenu.
    - Un jeu d'identifiants peut être utilisé pour plusieurs sources de contenu partageant la même identité source.

- Utilisez les LCA pour limiter la sécurité des documents. Cette méthode permet d'accélérer l'autorisation tout en optimisant les performances de recherche.

Annexe A

Exemple de code client Trusted Application écrit 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;

//spécifique à l'utilisateur 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)
      {
        //assume session timed out
        _gsaSessionId = null;
        iRetry++;
        goto Initiate;
```

```
            }
          else
            throw e; //Si l'échec se reproduit, il peut être dû à une autre cause.
        }

      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;
    }

  }
}
```