Vers une PKI globale basée sur DNSSEC et LDAP

Maryline Laurent-Maknavicius
CNRS Samovar UMR 5157, GET/INT/LOR, 9 rue Charles Fourier, 91011 Evry, France

Résumé

Cet article présente une infrastructure de gestion de clé (PKI) globale basée sur DNSSEC et LDAP. DNSSEC est actuellement le standard de l'IETF visant à protéger les enregistrements DNS et à publier des clés publiques associées à des noms de domaine. Si DNSSEC a l'avantage de mettre en relation de confiance des organisations privées, il ne permet pas de gérer des informations éphémères comme les clés publiques d'utilisateurs et de machines. C'est pourquoi, un deuxième niveau de PKI basé sur LDAP *(Lightweigth Directory Access Protocol)* est proposé pour gérer les clés publiques des utilisateurs, PC ou serveurs d'une organisation.

Cet article décrit la solution PKI proposée, donne des éléments sur l'implémentation et les tests réalisés et compare cette solution à d'autres travaux en cours comme les PKI traditionnelles.

Mots-clés: PKI, certificat électronique, LDAP, DNSSEC.

1 Les objectifs de la PKI globale

La plupart des communications sensibles passées sur Internet comme les échanges de messages électroniques, les transactions électroniques, et les connexions distantes sont protégées par des standards comme S/MIME, SSL/TLS, IPsec, et SSH. Ces standards reposent principalement sur des systèmes de cryptographie à clés publiques et souffre d'un problèmes de sécurité fondamental : comment avoir confiance dans l'association entre une clé publique et un propriétaire ? Cette association étant particulièrement critique pour authentifier des entités (utilisateurs, serveurs web, etc.), et pour assurer la confidentialité et l'intégrité des échanges, il y a un réel besoin de réguler la gestion des clés publiques.

Une infrastructure de gestion de clés ou PKI *(Public Key Infrastructure)* [Hous99] est responsable de l'organisation et de la mise en œuvre des aspects techniques lui permettant de générer et distribuer des clés privées et publiques aux propriétaires, de publier et révoquer des clés publiques, et de valider des clés publiques. En pratique, des tierce partie de confiance sont définies à cet effet. Elles sont organisées en une hiérarchie d'autorités de certification (AC) et chacune étant légalement habilitée à gérer des certificats électroniques.

Cet article décrit une solution techniquement simple qui résulte du projet GET CADDISC et qui est basée sur les annuaires DNSSEC et LDAP. Le but n'est pas de remplacer les fournisseurs de service de certification actuels (qui offre un service PKI de très bon niveau de sécurité), mais de fournir une approche PKI allégée en sécurité dans le but d'introduire un minimum de sécurité dans Internet. Par exemple, la solution présentée permettrait d'éviter des serveurs web sans certificats ou avec un certificat auto-signé.

Cet article nécessite un background technique sur la cryptographie à clés publiques et le DNS. Il est organisé comme suit. Le reste de la section 1 introduit les concepts de base des PKI, DNSSEC et LDAP. La section 2 décrit le couplage de DNSSEC et LDAP dans le but d'obtenir une PKI globale. La section 3 présente les aspects développement et tests et la section 4 les considérations liées au déploiement. La section 5 compare l'approche CADDISC aux travaux de recherche en cours. Une liste d'acronymes utiles est fournie en fin d'article.

- serialNumber: un identifiant unique attribué au certificat par l'AC
- issuer: le DN (Distinguished Name) de l'AC ayant généré le certificat
- issuerAltName *(issuer alternative name)* : une autre possibilité d'identifier l'émetteur à l'aide d'une adresse email, une adresse IPv4, une URI *(Uniform Resource Identifier)*, etc.
- validity: le début et fin de période de validité du certificat
- subject: le DN du propriétaire du certificat
- subjectAltName *(subject alternative name)* : un autre identifiant du propriétaire comme une adresse email, une adresse IPv4, une URI
- subjectPublicKeyInfo: la clé publique associée au propriétaire du certificat
- cRLDistributionPoints *(CRL distribution points extension)* : cet élément identifie comment les informations CRL peuvent être obtenues (e.g. URI)
- extKeyUsageSyntax *(extended key usage)* : le champ précisant les différentes utilisations qui peuvent être faites du certificat
- signatureValue: la signature générée sur le contenu du certificat par l'AC pour prouver la validité du certificat

Figure 1. **Extrait des certificats X.509 v3 (Housley et al., 2002).**

1.1 Certificats et liste de révocation de certificats (CRL)

Les certificats électroniques sont définis pour garantir qu'une clé publique est associée à une entité (utilisateur, serveur web…). A cet effet, il existe des autorités de certifications (Housley et al., 2002) qui sont chargées d'identifier l'entité avant de garantir sa validité en apposant une signature sur le contenu du certificat. Les Acs peuvent être organisées en une hiérarchie d'Acs avec une AC racine qui délègue la gestion de certificats auprès d'autres ACs.

Comme le montre la figure 1, le certificat X.509v3 actuellement standard (Housley et al, 2002) inclut des champs obligatoires comme serialNumber, subject, issuer, validity, subjectPublicKeyInfo et signatureValue. Grâce à la signature de l'AC, toute entité ayant confiance en la clé publique de l'AC peut obtenir un accès sécurisé à n'importe clés publiques gérées par l'AC. S'il existe plusieurs niveaux d'Acs, alors chaque AC doit avoir sa propre clé publique signée par l'AC de niveau supérieure. Comme l'AC racine n'est pas dotée d'AC de niveau supérieure, l'AC racine doit signer son propre certificat. Notez que les certificats auto-signés sont sensibles aux usurpations d'identité du fait que n'importe qui est à même de générer des certificats auto-signés avec dans le champ subject n'importe quel contenu, ceci grâce à des logiciels opensource. Toute la chaîne de ACs depuis le plus bas niveau d'AC forme une chaîne de certification qui sert à valider les clés publiques. L'idée est que la clé publique de l'AC racine est connue de telle sorte que la validation de la clé publique consiste à valider tous les certificats des Acs appartenant au chemin de certification.

Quand une clé publique est révoquée, son numéro de série doit petre ajouté à une liste de révocation de certificats CRL signée par l'AC. La procédure de validation d'une clé publique inclut la vérification que la clé publique n'est pas dans le CRL. Habituellement, les certificats incluent l'information de localisation de la CRL dans le champ cRLDistributionPoints.

1.2 DNSSEC

Le standard DNSSEC (Eastlake, 1999) définit des extensions de sécurité pour le DNS. Il assure l'intégrité et l'authentification des enregistrements DNS (RR - *Registration Record*) en introduisant des signatures électroniques (RRSIG RR) calculée par la zone sécurisée DNSSEC sur des enregistrements DNS. En fait, chaque zone possède deux paires de clés publiques/privées appelées Key Signing Key (KSK) et Zone Signing Key (ZSK). La clé ZSK signe les enregistrements gérés par la zone elle-même tandis que la clé KSK est la clé publique connue par la zone parent pour authentifier sa zone fille.

La zone parent certifie publiquement la clé publique de sa zone fille en signant un hash sur elle et en publiant ce hash dans l'enregistrement Delegation Signer (DS RR) (Gudmundsson, 2003) (cf. figure 2). La zone fille publie la clé publique KSK dans l'enregistrement contenant la clé publique (DNSKEY RR) et qui doit être auto-signée par la zone fille en utilisant la clé privée KSK. La clé ZSK est publiée par la zone fille dans DNSKEYRR signée par la clé privée KSK locale. Avec les notions de DS et DNSKEY RRs pour chaque zone sécurisée, il est possible de mapper une PKI sur la hiérarchie DNS avec chaque zone définissant un AC intermédiaire et un AC racine étant le DNS sécurisé le plus haut dans la hiérarchie. Cet article appelle une zone DNSSEC sécurisée comme une zone DNSSEC ayant un parent non DNSSEC.

La publication de clés publiques associées à des noms DNS ou des adresse IP peut prendre deux formes, l'usage des enregistrements DNSKEY RR protégés par les enregistrements RRSIG RR.

DNSSEC implémente deux possibilités pour publier des clés publiques associées à des nomes DNS ou des adresses IP. Tout d'abord la combinaison de DNSKEY RR et RRSIG RR permet à une zone de garantir l'association entre clés publiques associées et noms de domaine. La sécurité de l'association est entièrement assurée par la PKI DNS. Ensuite DNSSEC définit un CERT RR pour publier les certificats ou CRL associés à des noms de domaines. La sécurité de l'association est encore assurée par la PKI DNS grâce à RRSIG RR, mais les certificats ou CRLs sont gérés par une AC extérieure. Par exemple, dans le projet CADDISC, nous définissons un certificat auto-signé dans un CERT RR associé au nom de domaine "int-evry.caddisc.enst.idsa.prd.fr". Ce certificat correspond à l'AC racine de l'institut INT, et DNSSEC garantit sa validité. En effet, pendant la vérification, le resolver DNSSEC qui fait confiance à la zone "idsa.prd.fr" doit vérifier les clés publiques (ZSK et KSK) de la zone "enst.idsa.prd.fr" jusqu'à la zone "int-evry.caddisc.enst.idsa.prd.fr". Enfin, le resolver doit vérifier la signature qui protège le CERT RR.
DNSSEC n'est pas recommandé pour la gestion de CRL avec CERT RR, ni les certificats d'utilisateur avec DNSKEY RR ou CERT RR. Comme le cache DNS introduit des délais dans la prise en compte des mises à jours des enregistrements, DNSSEC n'apparaît pas souhaitable pour gérer les informations de courtes durée et en particulier à l'échelle des entreprises ou universités où les déplacements de personnes et de machines sont fréquents.

car ces informations sont de courte durée et le cache DNS introduit des délais dans la prise en compte des mises à jours des enregistrements, DNSSEC n'apparaît pas souhaitable pour gérer les informations de courtes durée et en particulier à l'échelle des entreprises ou universités où les déplacements de personnes et de machines sont fréquents.

pour la gestion de CRL avec CERT RR, ni les certificats d'utilisateur avec DNSKEY RR ou CERT RR. Comme le cache DNS introduit des délais dans la prise en compte des mises à jours des enregistrements, DNSSEC n'apparaît pas souhaitable pour gérer les informations de courtes durée et en particulier à l'échelle des entreprises ou universités où les déplacements de personnes et de machines sont fréquents.

Zone DNSSEC racine ("idsa.prd.fr. ")

enst.idsa.prd.fr:
- DS RR: Hash portant sur la clé publique KSK de la zone "enst.idsa.prd.fr"
- RRSIG RR: Signature calculée sur le DS RR avec la clé locale ZSK de "idsa.prd.fr"

Serveur DNSSEC pour la zone " enst.idsa.prd.fr "

- DNSKEY RR: clé publique KSK locale
- RRSIG RR: signature calculée avec la clé privée KSK sur le DNSKEY RR contenant KSK

- DNSKEY RR: clé publique locale ZSK
- RRSIG RR: signature calculée à l'aide de la clé privée ZSK et portant sur DNSKEY RR contenant ZSK

caddisc.enst.idsa.prd.fr:
- DS RR: Hash portant sur la clé publique KSK de la zone "caddisc.enst.idsa.prd.fr"
- RRSIG RR: Signature calculée sur le DS RR avec la clé locale ZSK de "enst.idsa.prd.fr"

Figure 2. **Enregistrements inclus dans la hiérarchie de serveurs DNSSEC.**

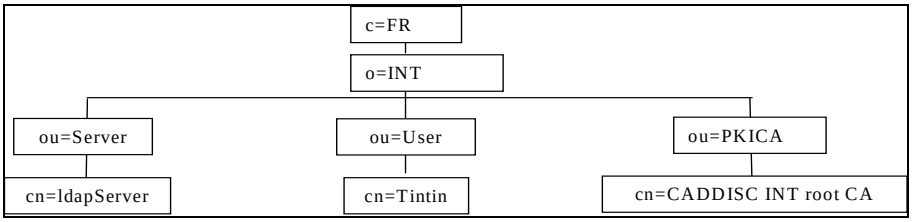

Figure 3. **Exemple de la base LDAP de l'INT.**

LDAP signifie Lightweight Directory Access Protocol (Wahl et al., 1997), et était à l'origine conçu pour XXXX
LDAP est un protocole standard ouvert pour accéder à distance à un annuaire où les données sont organisées en arbre (cf. figure 3). Habituellement le serveur LDAP publie des données associées à des machines, des utilisateurs, etc. comme des login, des noms, des mots de passe, des privilèges… Par conséquent, LDAP est adapté à la

publication de certificats et de CRL à l'échelle d'un réseau privé. Dans ce but, de nouveaux attributs (Boeyen et al., 1999a) sont définis comme pkiUser, pkiCA, userCertificate, cACertificate, certificateRevocationList, et authorityRevocationList ainsi que des méthodes vériées (Boeyen et al., 1999b) pour rechercher des certificats dans un serveur LDAP.

Le certificat associé à un utilisateur peut être aisément lu depuis un browser traditionnel (Netscape) ou un client LDAP (ldapbrowser) avec un identificateur URI *(Uniform Resource Identifier)* approprié comme : ldap://ldapServer.int-evry.fr/CN=Tintin,ou=User,o=INT,c=FR?UserCertificate

Cet URI signifie que le serveur LDAP nommé ldapServer du domaine int-evry.fr est interrogé pour retourner l'attribut LDAP UserCertificate pour l'utilisateur Tintin dont le DN est CN=Tintin,ou=User,o=INT,c=FR.

2 Combination of DNSSEC and LDAP

Une PKI globale peut bénéficier des capacités de LDAP et DNSSEC à gérer des clés publiques à la condition qu'une chaîne de certification de confiance soit définie entre DNSSEC et LDAP.

Pour illustrer, le reste de l'article suppose que l'institut INT dispose d'un nom de domaine officiel (int-evry.fr) et d'un nom de domaine spécifique au projet CADDISC (int-evry.caddisc.enst.idsa.prd.fr) qui intègre DNSSEC. L'INT possède un serveur LDAP nommé ldapServer où les certificats des utilisateurs et serveurs sont stockés et mis à la disposition du grand public.

2.1 La chaîne de confiance

La chaîne de confiance entre DNSSEC et LDAP est obtenue de trois façons différentes. Tout d'abord, DNSSEC sert à publier de façon sécurisée le certificat auto-signé de l'AC racine de l'INT nommé "CADDISC INT root CA". L'AC racine peut être considérée de confiance grâce à la PKI DNSSEC, comme l'explique la section 1.2. Ensuite, ce même certificat est disponible dans la base LDAP dans l'entrée correspondant au DN = "CN=CADDISC INT root CA,ou=PKICA, o=INT,c=FR". Enfin, le certificat auto-signé racine inclut à la fois une référence LDAP et une référence DNS dans les champs subjectAltName et issuerAltName (cf. figure 4) ce qui rend possible l'obtention de certificat soit depuis l'annuaire DNSSEC, soit depuis l'annuaire LDAP, ce qui est utile pendant la vérification de certificats.

Avec la hiérarchie d'AC à un seul niveau, le certificat de l'utilisateur Tintin est signé par l'AC racine elle même, si bien que le DN de l'AC racine et la référence LDAP sont fournis dans les champs issuer et issuerAltName. Le champ issuerAltName permet à n'importe quelle entité de localiser le serveur LDAP en charge de la publication du certificat de l'utilisateur Tintin et du certificat de l'émetteur. Remarquez que le champ subjectAltName correspond à l'adresse de messagerie électronique tandis que celui du ldapServer au nom du serveur.

2.2 La révocation par CRL

Si un certificat est révoqué, l'AC responsable de ce certificat doit générer une CRL incluant son numéro de série. Par exemple, la révocation du certificat de Tintin conduira à la génération d'une CRL par l'AC racine de l'INT et la publication de cette CRL dans la base LDAP. Pour vérifier le certificat, il y a besoin de spécifier dans le certificat (champ cRLDistributionPoints) une référence URI de la CRL courante (cf. figure 4).

La révocation du certificat de l'AC racine doit être gérée par le serveur DNS responsable et consiste simplement à supprimer le CERT RR comme aucune CRL n'est maintenue par le DNS.

Certificat de l'AC racine :

issuer: C=FR, O=INT, OU=PKICA, CN=CADDISC INT root CA
issuerAltName: DNS:int-evry.caddisc.enst.idsa.prd.fr, URI:ldap://ldapServer.int-evry.fr/CN=CADDISC INT root CA;ou=PKICA;o=INT;c=FR?cACertificate
subject: C=FR, O=INT, OU=PKICA, CN=CADDISC INT root CA
subjectAltName: DNS: int-evry.caddisc.enst.idsa.prd.fr, URI:ldap://ldapServer.int-evry.fr/CN=CADDISC INT root CA;ou=PKICA;o=INT;c=FR?cACertificate

Certificat de l'utilisateur Tintin

issuer: C=FR, O=INT, OU=PKICA, CN=CADDISC INT root CA
issuerAltName: URI:ldap://ldapServer.int-evry.fr/CN=CADDISC INT root CA;ou=PKICA;o=INT;c=FR?cACertificate
subject: C=FR, O=INT, OU=User, CN=Tintin
subjectAltName: email:Tintin@int-evry.fr
cRLDistributionPoint: URI:ldap://ldapServer.int-evry.fr/CN=CADDISC INT root CA;ou=PKICA;o=INT;c=FR?certificateRevocationList

Certificat du serveur ldapServer

issuer: C=FR, O=INT, OU=PKICA, CN=CADDISC INT root CA
issuerAltName: URI:ldap://ldapServer.int-evry.fr/CN=CADDISC INT root CA;ou=PKICA;o=INT;c=FR?cACertificate
subject: C=FR, O=INT, OU=Server, CN=ldapServer
subjectAltName: ldapServer.int-evry.fr
cRLDistributionPoint: URI:ldap://ldapServer.int-evry.fr/CN=CADDISC INT root CA;ou=PKICA;o=INT;c=FR?certificateRevocationList

Figure 4. **Certificats signés par l'AC racine de l'INT et stockés dans l'annuaire LDAP de l'INT.**

3 Certificate verification processing

La vérification de certificat doit être traité en deux étapes, la première pour télécharger les certificats appartenant à la chaîne de certification et la seconde pour vérifier la validité de chaque certificat de la chaîne. Plus précisément :
1 - Tous les certificats appartenant à la chaîne de certification sont téléchargés depuis l'AC de niveau le plus bas jusqu'à l'AC de plus haut niveau dans la hiérarchie PKI, ceci en se basant sur l'information issuerAltName dans les certificats. Pour le certificat de Tintin, le serveur LDAP sera sollicité une fois pour le certificat de l'AC racine car la hiérarchie PKI de l'INT est à un niveau. Dès que le certificat racine est trouvé dans la chaîne, la hiérarchie DNS est sollicitée pour fournir le CERT RR correspondant à la référence DNS donnée dans le champ issuerAltName ou subjectAltName du certificat racine. La validité du CERT RR retourné est assuré par la hiérarchie DNS (cf. section 1.2), mais le certificat sera définitivement considéré valide si le même certificat est obtenu depuis la requête LDAP.

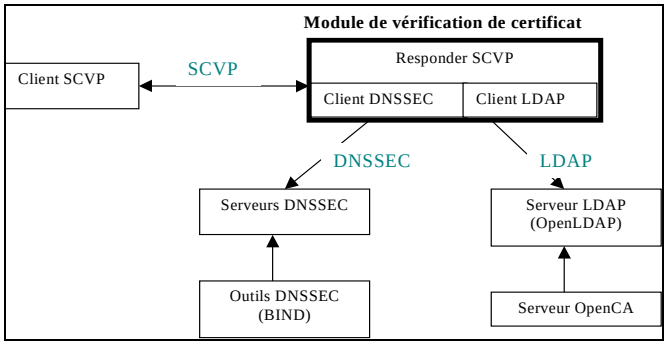

Figure 5. **Architecture de vérification de certificat.**

1 - Tous les certificats téléchargés sont vérifiés en prenant en compte leur date de validité (validity), la signature (signatureValue) et, si possible, une CRL. La vérification est faite depuis le certificat de niveau le plus élevé jusqu'au niveau le plus bas. Pour la vérification de la révocation, il est nécessaire de téklécharger la CRL correspondant à l'URI CrlDistributionPoints du certificat en cours de vérification et de vérifier sa propre période de validité et signature.

4 Certificate verification architecture

Pour éviter que les applications ne prennent en charge la vérification de certificats, l'architecture CADDISC définit un module de vérification centralisé. Actuellement, le protocole OCSP (Online Certificate Status Protocol) (Myers et al., 1999) est le seul standard IETF disponible pour les applications comme Netscape 7, et OpenSSL 0.9.7, mais il ne satisfait pas complètement nos besoins car OCSP est conçu pour faire de la validation de certificat seulement. C'est-à-dire, un serveur OCSP (OCSP responder) valide un certificat par rapport à une CRL, mais ne vérifie pas la chaîne de certification et en temps que tel propose des types de statut limités pour qualifier un certificat. Dans le but de vérification, l'IETF est en train de définir un nouveau protocole appelé SCVP (Malpani et al., 2003) pour Siumple Certificate Validation Protocol qui sera très probablement intégré dans de futures applications nécessitant des PKI (client de messagerie, navigateurs…).
Comme la chaîne de certification inclut des références DNS et LDAP, le logiciel de vérification de certificat doit nécessairement intégrer un client DNS et un client LDAP, comme le montre la figure 5. C'est-à-dire, un client SCVP qui a été développé dans le projet CADDISC fait une requête pour vérifier que le certificat est dans son module de vérification local, et le module est responsable de la récupération de tous les certificats de la chaîne pour vérification. Le client DNSSEC est chargé d'obtenir les enregistrements CERT, DNSSKEY, et RRSIG des serveurs DNS et doit être configuré avec la clé publique de la zone DNSSEC racine "idsa.prd.fr" (fichier de configuration /etc/sresolv.conf) pour vérifier la signature DNSSEC. Le serveur LDAP est le gestionnaire du certificat sous vérification et peut être distant du responder SCVP.

De plus, les communications avec le serveur LDAP et les serveurs DNSSEC ne sont pas nécessairement sécurisées car les certificats retournés sont déjà signés, mais le point le plus critique est la communication entre le client SCVP et le responder SCVP comme le client doit être sûr de l'identité du responder pour ne pas être abusé par un usurpateur.

5 Implementation and testing

Nous avons obtenu une implémentation dans les environnements Mandrake 9 et FreeBSD 4.7/4. Comme l'implémentation est opensource, nous avons sélectionné OpenLDAP2.1.22 pour implémenter le serveur LDAP, et BIND 9.3.0 *(Berkeley Internet Name Domain)* (Albitz and Liu, 2002) pour le serveur DNS. Comme aucun logiciel pour SCVP n'existait, l'ENST-Bretagne a implémenté le client et serveur SCVP.
Pour générer des certificats et CRL, nous avons sélectionné OpenCA0.9.1 avec Apache1.3.27 et OpenSSL0.9.7 pour son interopérabilité avec OpenLDAP fournissant ainsi l'opportunité de publier automatiquement les certificats et CRL dans la base LDAP. La difficulté était de rendre les DN cohérents entre OpenLDAP et OpenCA du fait que les propriétaires du certificat est identifié avec un DN et doit être stocké dans l'entrée LDAP correspondante. Une autre difficulté a été de générer des certificats avec des champs appropriés spécifiques au projet CADDISC comme subjectAltName, et issuerAltName. Dans ce but, nous avons modifié la configuration des fichiers OpenSSL comme openssl.conf, User.conf, User.ext, Web_Server.conf, Web_Server.ext… Comme le montre la figure 4, le certificat de ldapServer inclut l'URI de son propre certificat dans subjectAltName et l'URI du certificat de l'AC racine dans issuerAltName.
Le module de vérification de certificat étend la fonction verify de OpenSSL et nécessite le développement d'une nouvelle méthode lookup pour OpenSSL. Plus précisément, la méthode trust est extraite du validateur DNSSEC du projet IDsA (DNSToolKit library) (IDsA, 2004) de telle sorte que les certificats vérifiés par un client DNS peut être considéré comme de confiance par OpenSSL. La méthode lookup de LDAP a été définie pour obtenir des certificats depuis les serveurs LDAP.

Deux tests sont décrits ci-dessous, l'un porte sur un certificat d'utilisateur local et l'autre sur un certificat d'utilisateur distant.

5.1 Test d'un certificat d'utilisateur local

Comme le montre la figure 6, le responder SCVP est tout d'abord lancé sur le port 8000 et le client SCVP fait une requête sur le serveur pour vérifier le certificat TintinCert.der. L'information retournée au client est **"cr=0"** ce qui signifie que le certificat est considéré comme valide. Le commentaire donné sur le serveur "**trust_dnnsec: FQDN=int-evry.caddisc.enst.idsa.prd.fr"** signifie que la vérification DNS pour le certificat de l'AC a bien marché.

5.2 Test d'un certificat d'un utilisateur distant

Supposons qu'un utilisateur d'une entreprise externe a besoin de vérifier le certificat de Tintin comme le décrit la figura 7. Son responder SCVP doit interroger le serveur LDAP de l'INT **(ldapServer.int-evry.fr)** pour récupérer le certificat de l'AC de l'INT (première recherche LDAP de la figure 7). Ensuite, il doit récupérer auprès des serveurs DNSSEC le même certificat de l'AC et le comparer les deux. Enfin, le responder SCVP doit confronter le certificat de Tintin à sa propre CRL en demandant au serveur ldapServer de l'INT de lui fournir la CRL (seconde recherche LDAP de la figure 7). Le certificat de Tintin est considéré valide avec la réponse **"cr=0"**.

6 Deployment considerations

6.1 DNSSEC

L'efficacité de la PKI globale proposée est étroitement liée au déploiement du DNSSEC. Aujourdh'ui, DNSSEC est toujours expérimental et quelques administrateurs de zones son réticents à l'idée de l'introduire car il est gourmand en volume (environ sept fois plus volumineux) et beaucoup plus complexe à gérer. Une implémentation du service DNSSEC sécurisé suppose que les clés privées sont stockées dans un système déconnecté du réseau, la signature de zone est traitée dans une machine séparée, l'information sur la clé est échangée avec la zone parent et les clés sont régulièrement mises à jour.
Le but idéal de DNSSEC serait de mapper une PKI sur la hiérarchie DNS. Cela signifie de configurer le serveur racine DNS (connu sous ".") comme la zone DNSSEC racine (cf. section 1.2), de telle sorte que la chaîne de certification de confiance inclut toujours le certificat de la zone DNSSEC racine. Dans l'implémentation, cela résulterait en le client DNSSEC configuré avec une seule clé publique, celle du serveur DNSSEC racine.
Aujourd'hui, DNSSEC est loin de cette perspective comme il existe de nombreuses îles DNSSEC sécurisée dans la hiérarchie DNS et que ce n'est pas aussi sûr et pratique que cela pourrait l'être. Chaque île DNSSEC est une PKI indépendant et comme les Pkis traditionnels (voire section 5.1), il y a un besoin faire confiance à chaque PKI séparément. Dans l'implémentation, ceci est implémenté dans le client DNSSEC qui enregistre toutes les clés publiques des serveurs DNSSEC racine.

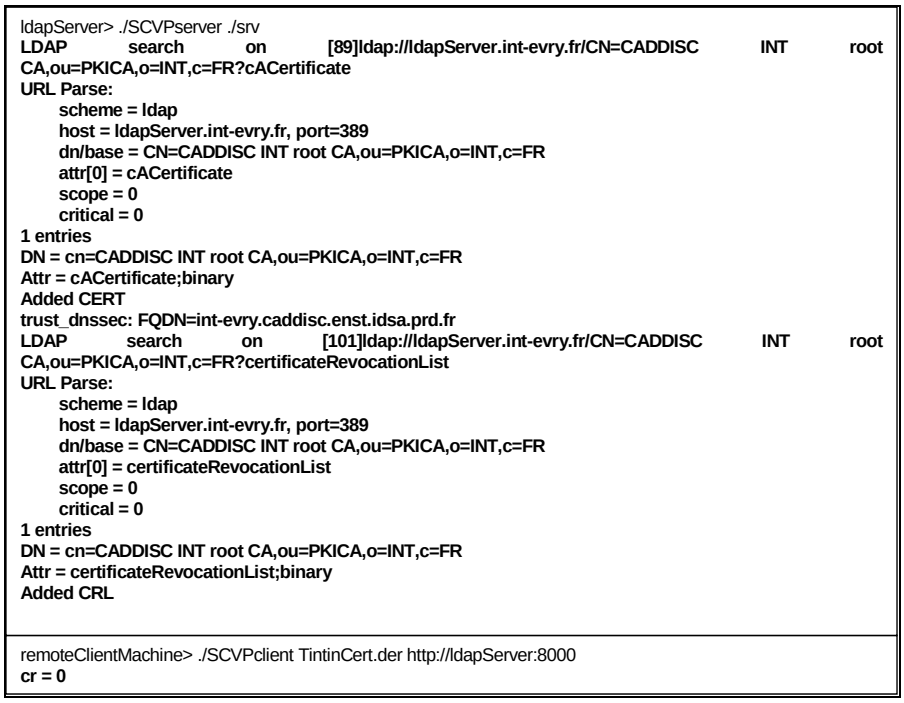

Figure 7. **Test d'un certificat d'utilisateur distant.**

ldapServer> ./SCVPserver ./srv **trust_dnssec: FQDN=int-evry.caddisc.enst.idsa.prd.fr**
ClientMachine> ./SCVPclient TintinCert.der http://ldapServer:8000 **cr = 0**

Figure 6. **Test d'un certificat d'utilisateur local.**

Ces dernières années, XXXXX
During the past few years, we noted collaborative international efforts towards more security in DNS. One international DNSSEC experiment is currently being realized within the OTDR testbed (Operational Testbed for new DNS features in the Internet Root) to test if DNSSEC is ready for deployment. Some DNSSEC shadow zones are already operational, and applied to a pretty large scale as they are synchronized with real non-DNSSEC zones like ".fr", ".nl"… Those shadow zones are not yet referenced in the DNS root server, but are referenced in the OTDR root server. Considering those strong efforts, it is likely that DNSSEC will be gradually deployed.

6.2 Eviter le spam

Dans LDAP et OpenLDAP, une opération de recherche peut être effectuée sur un sous-arbre de la base LDAP. Par exemple, l'URI suivante avec le scope égal à "sub" (pour "subtree") retournera tous les certificats des employés ou machines disponibles dans la base LDAP : **ldap://ldapServer.int-evry.fr/o=INT, c=fr?userCertificate?sub**
Ainsi, il y a un risque que des personnes malveillantes téléchargent la liste complète des certificats avec leurs noms et adresses de messagerie, et par exemple réalisent des attaques de virus au travers de l'application de messagerie ou du spam. Pour empêcher ces attaques, le serveur LDAP doit être configuré avec l'option limits mise à 1 de telle sorte que les personnes anonymes ne puisse accéder qu'à une seule entrée à la fois. Ceci est configuré dans le fichier ldap.conf avec la commande suivante : **limits anonymous size=1**.
Si le serveur LDAP de l'entreprise inclut des informations sensibles sur les employés, deux serveurs LDAP peuvent être installés, l'un privé comprenant la base de données complète de l'entreprise, l'autre publique limité aux informations publiques. Le serveur LDAP public peut être implémenté comme un serveur LDAP dupliqué avec une copie partielle des données de la base privée.

6.3 Support de SCVP

Pour une utilisation automatique et simple de la PKI globale, les applications doivent supporter SCVP. Aujourd'hui, SCVP est toujours en cours de définition à l'IETF et le protocole OCSP est intégré dans quelques applications. Dans le futur, comme SCVP offre la possibilité pour les applications de vérifier les chaînes de certificat, il est probable que SCVP et OCSP seront supportés dans les applications clientes.

6.4 Gestion de clés

Si un utilisateur dispose de plusieurs certificats, un pour l'authentification, un autre pour le chiffrement…, l'implémentation de CADDISC peut être aisément adaptée en définissant un DN pour chaque type de certificat. Par exemple, on pourrait définir le DN "cn=encryption,cn=Tintin,ou=User,o=INT,c=FR" pour le certificat à usage de chiffrement, le DN "cn=authentication,cn=Tintin,ou=User, o=INT,c=FR" pour le certificat à usage d'authentification. Cela ne résulterait pas dans des modifications fondamentales.
Dans la mise à jour de clés, plusieurs clés peuvent être valides simultanément pour un même usage. Par exemple, les ACs racine peuvent avoir deux clés publiques/privées stockées dans le serveur DNSSEC, une nouvelle pour générer de nouveaux certificats d'utilisateurs, et un ancien pour permettre la vérification des certificats d'utilisateurs générés avec l'ancienne clé privée de l'AC. En fait, les clés de l'AC doivent être mises à jour selon la durée de vie des certificats générés. Par exemple, si la durée de vie des certificats gérés est de un an, cela signifie que les clés des Acs doivent être mises à jour au moins un an avant leur expiration. Ainsi, une fois la clé de l'AC expirée, tous les certificats valides gérés par l'AC seront signés avec la nouvelle clé privée de l'AC.
Si les clés des utilisateurs sont mises à jour, plusieurs clés peuvent être valides simultanément et stockées dans le serveur LDAP. L'implémentation actuelle de CADDISC ne supporte pas le stockage de plus d'une clé dans les serveurs LDAP et DNSSEC, mais ceci peut être amélioré.
Après la compromission d'une clé privée, la révocation d'une clé d'AC conduit à la suppression de la clé publique de l'AC dans le derveur DNSSEC tandis que la révocation de clés ordinaires résulte dans la génération et publication d'une nouvelle CRL dans le serveur LDAP. La révocation de la clé publique de l'AC n'est pas immédiate à cause du cache DNS tandis que la révocation des clés d'utilisateur est immédiate grâce à la publication de CRL via LDAP.

7 Comparaison avec les solutions alternatives

Cette section compare l'approche de CADDISC avec les PKI traditionnels, les solutions de bridge-CA et de certificat croisés, et la solution (Wheeler, 2002) basée sur LDAP et DNSSEC.

7.1 PKI traditionnel

Les PKI traditionnels consistent à définir une hiérarchie de CAs (Housley et al., 1999), qui est gérée par l'entreprise elle-même, par un fournisseur de service de certification (CSP) ou les deux. Si la PKI est implémentée dans les entreprises elles-même, les logiciels open source comme OpenCA peuvent être utilisés pour délivrer des certificats aux employés et machines.Les avantages sont le faible coût et l'impression de sécurité donnée par le contrôl totale de la PKI. L'inconvénient est comment établir des relations de confiance entre les entreprises externes, c'est-à-dire comment leur prouver la validité du certificat de l'AC racine.
L'approche outsource offre un meilleur niveau de sécurité car les certificats sont gérés avec des procédures très strictes et en particulier la clé privée de l'AC racine est sur protégée. L'inconvénient est le coût des certificats et le manque de flexibilité du à l'entreprise qui ne contrôle plus les certificats. L'avantage est que les CSPs sont des sociétés publiques et que les certificats des CSP peuvent être plus facilement de confiance par des entités extérieures. Par exemple, si le certificat de l'AC racine est préenregistré dans les navigateurs web, et localement considérés comme de confiance, un client peut accepter des certificats de façon sécurisée. Si le CSP n'est pas enregistré, le problème reste le même que dans la première approche. Il n'y a de garantie que le certificat chargé soit attaché à l'entité déclarée, et la décision de faire confiance ou pas au certificat revient à l'utilisateur. Ainsi, le danger est grand que des utilisateurs se fassent passer pour d'autres utilisateurs en définissant un AC factice avec un certificat d'utilisateur incluant les caractéristiques de l'utilisateur à usurper. Pour empêcher cela, les certificats croisés et les bridge-CAs ont été définis comme le présente la section 5.2.
L'approche CSP offre une gestion de clés plus sûre que l'approche CADDISC. En fait, les deux solutions répondent à différents besoins de sécurité. L'approche CSP est intéressante pour les entités qui ont de gros besoin en sécurité et peuvent inclure une assurance assurant des indemnités en cas de pertes financières, tandis que la solution CADDISC vise les entreprises qui ne peuvent pas se permettre de se payer un service d'enregistrement de certificat et ont l'habitude de définir pour leur serveur des certificats auti-signés.

7.2 Certificat croisé ou bridge-CA

Pour améliorer les PKI traditionnels, et prévenir les risques d'usurpation d'AC (cf. section 5.1), les Acs racine peuvent convenir de générer des certificats croisés pour les autres ACs racines (Housley et al., 1999), c'est-à-dire des certificats certifiant l'association entre l'identité d'un AC et d'une clé publique. Le certificat croissé est un accord passé entre une paire d'ACs qui doit être répété pour chaque AC de confiance ; les certificats croisés ne peuvent pas être établis dynamiquement comme ils supposent un accord entre PKIS ; ils ne sont pas déployés aujourdh'ui.
Le bridge-CA est un CA racine indépendant en haut des Acs enregistrés appartenant aux PKIs indépendantes. Une PKI doit enregistrer sa clé publique auprès du bridge-CA pour que les entités ayant confiance dans le bridge-CA fassent confiance l'AC de la PKI. Aujourdh'ui le déploiement de bridge-CA est limité à des profils d'AC spécifiques et est déployé dans une zone géographie limitée, de telle sorte que le problème de confiance reste le même mais au niveau du bridge-CA.
En fait, définir un bridge-CA de niveau mondial est techniquement possible (avec des serveurs dupliqués) mais apparaît politiquement impossible car cela confèrerait au bridge-CA beaucoup trop de pouvoir et le problème de choisir l'entreprise ou l'organisation en charge de la gestion du bridge-CA s'avèrerait critique. L'approcha CADDISC repose sur des technologies existantes DNS et LDAP, et cela ne requiert pas beaucoup d'effort pour le déploiement et l'organisation.

7.3 Solution basée sur LDAP/DNSSEC

L'approche CADDISC est assez similaire à cette solution décrite dans (Wheeler, 2002), mais (Wheeler, 2002) se focalise sur les aspects gestion et déploiement et ne présente pas de détails d'implémentation comme la définition de la chaîne de confiance entre LDAP et DNSSEC, ni le gestion de CRL. De plus, (Wheeler, 2002) ne considère pas l'architectire de vérification de certificat centralisée de CADDISC, mais préconise une vérification distribuée où les applications vérifient elles-même les certificats avant de les utiliser. Comme les applications peuvent nécessiter la vérification de certificats et la vérification de certificat est consommatrice de CPU, l'approche centralisée apparaît meilleure.

8 Conclusions

Cet article présente une solution de PKI globale basée sur les technologies DNSSEC et LDAP. Elle a été implémentée et testée dans le projet GET CADDISC. Les détails techniques sont donnés comme la chaîne de confiance entre DNSSEC et LDAP, le traitement de la vérification de certificats, la gestion de la révocation de certificats, l'architecture logicielle, et les résultats de tests. Pour de plus amples détails, les lecteurs peuvent se référer à la page web de CADDISC à l'URL (CADDISC, 2004).

L'approche de PKI globale est complémentaire des PKIs traditionnels. Ces dernières sont considérées comme les PKI les plus sûres qui peuvent être améliorées avec la mise en œuvre de certificats croisés et de ridge-CAs. D'autre part, l'approche PKI globale est une approche légère en sécurité qui convient bien aux petits budgets et peut être considéré comme la PKI de niveau de sécurité par défaut. L'avantage de cette PKI globale est que les services DNS et LDAP sont déjà opérationnels et nécessite donc un minimum d'efforts pour installer et utiliser. De plus, si le protocole de validation de certificats (SCVP) est intégré dans les applications futures (clients de messagerie, navigateurs…), alors il sera possible de vérifier automatiquement et simplement des certificats avant de les utiliser.

Dans les prochains mois, les résultats de CADDISC seront intégrés dans les trois environnements suivants dans le projet GET VERICERT (VERICERT, 2004): l'authentification de mobiles dans les réseaux IEEE 802.11, l'authentification d'utilisateurs au réseau d'accès de l'opérateur (basé sur PANA - *Protocol for carrying Authentication for Network Access*), et l'authentification d'équipements IPsec.

REMERCIEMENTS

Francis + GET

ACRONYMS

AC Autorité de Certification	PKI Public Key Infrastructure
CRL Certificate Revocation List	RR Resource Record
CSP Certificate Service Provider	SCVP Simple Certificate Validation Protocol
OCSP Online Certificate Status Protocol	URI Uniform Resource Identifier

REMERCIEMENTS

Je tiens à remercier le GET (Groupe des Ecoles des Télécommunications) pour le soutien financier apporté aux projets CADDISC et VERICERT. Je tiens de plus à remercier les collaborateurs de ces projets, et en particulier Mr Francis Dupont pour avoir XXXXXXXXX

, et en particulier Mr Francis Dupont pour les développements

REFERENCES

Housley, R., Polk, W. , Ford, W., and Solo, D., 2002, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, RFC 3280.

Housley, R., Ford, W., Polk, W., and Solo, D., 1999, *Internet X.509 Public Key Infrastructure: Certificate and CRL Profile*, RFC 2459.

Eastlake, D., March 1999, *Domain Name System Security Extensions*, RFC 2535.

Gudmundsson, O., 2003, *Delegation Signer (DS) Resource Record (RR)*, RFC 3658.

Wahl, M., Howes, T., and Kille, S., December 1997, *Lightweight Directory Access Protocol (v3)*, RFC 2251.

Boeyen, S., Howes, T., and Richard, P., 1999a, *Internet X.509 Public Key Infrastructure: LDAPv2 Schema*, RFC 2587.

Howes, T. and Smith, M. , 1997, *The LDAP URL Format*, RFC 2255.

Boeyen, S., Howes, T., and Richard, P., 1999b, *Internet X.509 Public Key Infrastructure Operational Protocols - LDAPv2*, RFC 2559.

Myers, M., Ankney, R. , Malpani, A., Galperin, S., and Adams, C., 1999, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*, RFC 2560.

Malpani, A., Housley, R., and Freeman, T., 2003, *Simple Certificate Validation Protocol (SCVP)*, Internet Draft, draft-ietf-pkix-scvp-13.

Albitz, P., and Liu, C. , 2002, *DNS et Bind*, 4th edition, O'Reilly, ISBN: 2-84177-150-4.
IDsA, Infrastructure DNSSEC et Applications, 2004, http://www.telecom.gouv.fr/rnrt/projets/res_02_22.htm.
Wheeler, D.A., 2002, *Easier Email Security is on the Way?*, http://www.dwheeler.com/essays/easy-email-sec.html.
CADDISC GET project, 2004, web page http://www-lor.int-evry.fr/~maknavic/CADDISC/
VERICERT GET project, 2004, web page http://www-lor.int-evry.fr/~maknavic/VERICERT/