8

Protection des accès distants

Les accès distants au réseau d'entreprise offrent de nombreuses possibilités de pénétration. Les faiblesses de sécurité classiques reposent à la fois sur des lacunes d'authentification des utilisateurs et sur des failles des protocoles utilisés pour ces accès distants.

Ce chapitre traite de ces deux problèmes, authentification et protocoles, et détaille leurs solutions techniques.

La gestion des secrets associés aux accès distants reste le point de faiblesse principal en matière de sécurité réseau. La mise en place de procédures d'autorisation et de vérifications périodiques des droits d'accès des utilisateurs garantit la consistance de la base de données d'authentification et des droits d'accès des utilisateurs.

Assurer l'authentification des connexions distantes

Le laxisme qui entoure la gestion des secrets et des moyens d'authentification des accès distants a des répercutions de sécurité non négligeables sur l'entreprise. La plupart des accès distants se font à l'aide d'un ordinateur portable, qui ne contient généralement ni antivirus, ni pare-feu logiciel pour protéger les connexions des pirates qui scannent en permanence les plages d'adresses IP des opérateurs de télécommunications. Il s'ensuit que les moyens d'accès et d'authentification au réseau d'entreprise sont disponibles en libre-service.

De surcroît, des virus informatiques ont été spécialement développés pour rechercher sur des systèmes donnés tels que les PC portables tous les secrets relatifs aux accès distants.

L'authentification assure une protection contre toutes les attaques utilisant une usurpation d'identité, telles les attaques de type IP spoofing, qui simulent une adresse IP qui n'est pas celle de l'attaquant, les attaques visant à dérober les mots de passe, les attaques

par cheval de Troie, dont l'objectif est d'offrir à l'attaquant un accès non authentifié ou dérobé, les attaques visant à déchiffrer les mots de passe d'un système et les attaques utilisant des faiblesses de codage ou de protocoles d'authentification.

Règles de sécurité pour l'authentification des connexions distantes

Les règles de sécurité à considérer pour l'authentification des connexions distantes sont les suivantes :

- Tous les utilisateurs de l'entreprise sont connus et associés à une matrice de droits d'accès aux ressources de l'entreprise.
- Tous les accès au réseau d'entreprise (intranet) sont authentifiés. Cela concerne les accès des utilisateurs au réseau interne de l'entreprise aussi bien qu'aux ressources informatiques.
- Les accès distants au réseau d'entreprise (intranet) sont fortement authentifiés.
- Les connexions de tierces parties ou de fournisseurs du réseau d'entreprise (extranet) sont authentifiées. Aucune connexion directe au réseau interne de l'entreprise (intranet) n'est autorisée.

Cette section traite des solutions à mettre en œuvre afin d'offrir des services d'authentification des connexions distantes et détaille leurs aspects techniques.

Mots de passe

Le mot de passe est le schéma d'authentification le plus utilisé au monde. Simple, ne demandant aucun outil ou système de sécurité supplémentaire, il reste aussi le système le plus faible.

Les faiblesses des mots de passe viennent avant tout du fait que les protocoles d'accès usuels ne les chiffrent pas sur le réseau (Telnet, etc.). En second lieu, les mots de passe sont souvent mal choisis, et il est facile de les deviner à partir d'attaques sur les dictionnaires de mots de passe. Enfin, il s'agit d'une authentification de l'identité de l'utilisateur faible en soi, comparée à une authentification fondée sur un certificat électronique.

La seule protection efficace d'une authentification par mot de passe réside dans la qualité du mot de passe, lequel doit être généré de manière aléatoire. Il existe de bons outils pour cela, comme Password Safe, de Bruce Schneier, qui peut à la fois générer des mots de passe et les stocker sur son ordinateur personnel.

Password Safe chiffre une base de données de mots de passe à partir d'un mot de passe maître — le seul à retenir pour l'utilisateur — et génère les mots de passe automatiquement et de manière aléatoire, sans qu'il soit nécessaire de les mémoriser.

Tokens RSA

Depuis leur premier exemplaire, en 1986, les tokens RSA SecurID ont atteint en 1996 le million d'unités vendues. Cette technologie primée à de nombreuses reprises continue de dominer le marché de l'authentification des accès distants. Selon IDC, RSA Security détient 72 % de parts de marché des solutions d'authentification matérielle et logicielle.

La société a été couronnée de succès pour le déploiement à grande échelle des produits RSA SecurID et RSA ACE/Server.

Cette méthode repose sur la technologie des tickets, ou mots de passe valables pour une courte durée, environ soixante secondes. Lorsqu'un utilisateur veut se connecter au réseau, il se sert d'un authentifiant — token, ou carte à puce — et d'un code PIN secret. L'authentifiant génère alors des codes d'identification aléatoires toutes les soixante secondes grâce à un puissant algorithme. L'identification de l'utilisateur est effectuée en combinant le code PIN, l'authentifiant et le code aléatoire.

De cette manière, le code d'identification n'est valable qu'à un moment précis pour un utilisateur donné, ce qui rend impossible le vol et la réutilisation des mots de passe ou la découverte des mots de passe par attaque de dictionnaire.

Un serveur RSA ACE se charge d'administrer et de contrôler la validité des codes d'authentification de manière transparente pour l'utilisateur. Il existe de nombreuses formes d'authentifiants, que ce soit dans le domaine des tokens ou des cartes à puce. Les tokens sont des identifiants propres à chaque utilisateur.

Les tokens existent sous forme hardware ou software. Sous forme hardware, ils peuvent prendre la forme d'une calculatrice, d'une carte, d'un porte-clés, etc. Le token permet d'obtenir auprès du serveur RSA ACE un code d'authentification unique, différent à chaque connexion. Le code PIN permet d'activer le token, et le token de s'authentifier. Ces tokens ne nécessitent aucune installation particulière de logiciels sur les postes.

Sous forme software, le code d'authentification aléatoire est généré par un logiciel sécurisé, et non par un objet que l'on possède physiquement.

On parle alors d'authentification forte, car le token est possédé par l'utilisateur et le PIN est connu de l'utilisateur.

Signature numérique à paires de clés publique/privée

Avant de détailler comment réaliser une signature numérique, rappelons brièvement le fonctionnement des algorithmes de chiffrement à clé publique.

Les algorithmes cryptographiques à clés publiques, ou asymétriques, sont les algorithmes les plus utilisés de nos jours pour échanger des clés de chiffrement de session et pour la signature électronique. À l'inverse des algorithmes cryptographiques à clé secrète, ou symétrique, deux clés sont générées pour chaque utilisateur (privée, publique). Ces clés sont calculées à partir de règles précises, fondées sur la théorie des nombres. Nous verrons par la suite un exemple de calcul de biclé.

La figure 8.1 illustre la paire de clés publique/privée que possède John.

La clé publique peut être diffusée alors que la clé privée doit être soigneusement protégée. Si John souhaite envoyer un message à Joe, il chiffre le message avec la clé publique de Joe. De la sorte, seul Joe peut déchiffrer le message qui lui est destiné à l'aide de sa clé privée *(voir figure 8.2)*.

Figure 8.1
Paire de clés publique/ privée (biclé)

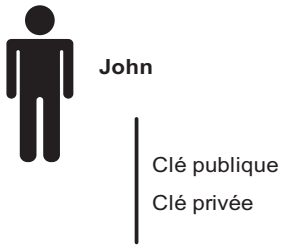
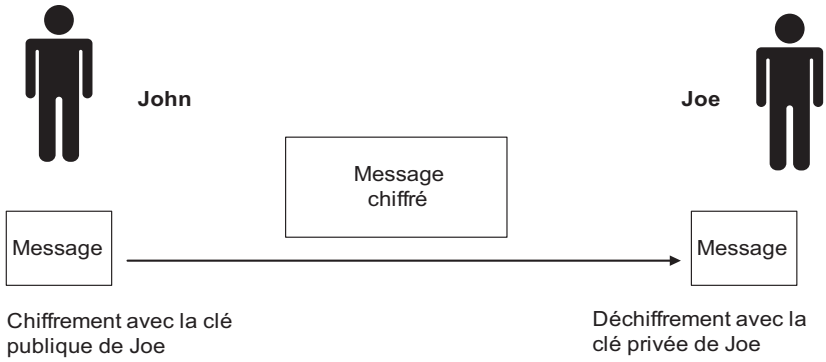
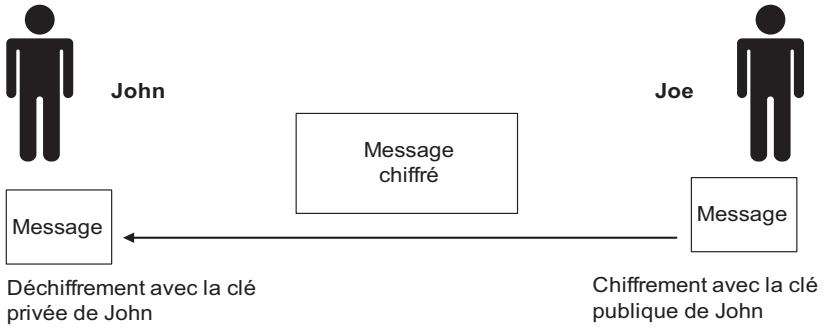


Figure 8.2
John envoie un message à Joe



Si Joe décide d'envoyer un message à John, il chiffre le message avec la clé publique de John, de sorte que seul John puisse le déchiffrer à l'aide de sa clé privée *(voir figure 8.3)*.

Figure 8.3
Joe envoie un message à John



Les algorithmes de chiffrement asymétrique ne font jamais transiter les clés privées sur le réseau. La sécurité de tels algorithmes tient au fait que la clé privée n'est pas divulguée et que, même avec les clés publiques, il est très difficile dans un temps raisonnable de calculer les clés privées à partir des clés publiques.

La génération des clés publique/privée suit des règles précises, fondées sur la théorie des nombres, et plus précisément des nombres premiers. L'espace des clés, c'est-à-dire l'ensemble des clés possibles, repose sur l'espace des nombres premiers utilisés dans les clés générées.

À titre d'exemple, nous allons réaliser le chiffrement et le déchiffrement d'un nombre par l'algorithme RSA.

Génération des clés

Soit deux nombres premiers, $p = 47$ et $q = 71$.

Le produit des deux nombres est le suivant : $p \times q = 47 \times 71 = 3\ 337$.

Dans la clé publique, *(e,n), e* est un nombre premier par rapport à :

$(p - 1) \times (q - 1) = (47 - 1) \times (71 - 1) = 3\ 220$

Prenons $e = 79$ de manière aléatoire, et vérifions que 79 est premier par rapport à 3 220 en calculant le PGCD(3 220, 79) à l'aide de l'algorithme d'Euclide (soit *a* et *b*, calculons PGCD(*a,b*) : $a = bq_0 + r_0$, $b = r_0q_1 + r_1$, …, $r_{n-1} = r_nq_{n+1} + r_{n+1}$, avec $r_{n+1} = 0$, alors PGCD(*a,b*) = r_n) :

(1) $3\ 220 = 79 \times 40 + 60$

(2) $79 = 60 \times 1 + 19$

(3) $60 = 19 \times 3 + 3$

(4) $19 = 3 \times 6 + 1$

79 est donc premier par rapport 3 220.

La clé privée *(d,n)* est calculée à partir de la formule suivante :

$d = e^{-1} \bmod[(p - 1)(q - 1)] = 79^{-1} \bmod(3\ 220) = 1\ 019$ (relation de Bezout : si *a* est inversible dans *Z/nZ,* il existe *u* et *v* tels que $a \times u + n \times v = 1$).

Si nous partons de la division euclidienne précédente, nous pouvons construire la relation de Bezout de la façon suivante :

(4) $19 - 3 \times 6 = 1$

(3) $3 = 60 - 19 \times 3$

(4) Combiné avec (3) : $19 - (60 - 19 \times 3) \times 6 = 1$

(4) $- 60 \times 6 + 19 \times 19 = 1$

(2) $79 - 60 \times 1 = 19$

(4) Combiné avec (2) : $- 60 \times 6 + (79 - 60 \times 1) \times 19 = 1$

(4) $- 60 \times 25 + 79 \times 19 = 1$

(1) $3\ 220 - 79 \times 40 = 60$

(4) Combiné avec (1) : $- (3\ 220 - 79 \times 40) \times 25 + 79 \times 19 = 1$

(4) $- 3\ 220 \times 25 + 79 \times (40 \times 25 + 19) = 1$

(4) $- 3\ 220 \times 25 + 79 \times 1\ 019 = 1$

1 019 est donc bien l'inverse de 79 dans $Z/(p - 1)(q - 1)Z$

Chiffrement/déchiffrement

Les calculs sont cette fois effectués dans le groupe multiplicatif $(Z/nZ,*)$.

Si nous désirons chiffrer le nombre *m,* nous appliquons la formule suivante avec la clé *(e,n) :* $c = m^e \bmod(n)$.

Pour $m = 688$, nous obtenons :

$c = 688^{79} \bmod(3\ 337) = 1\ 570$.

À l'inverse, si nous désirons déchiffrer *c,* nous appliquons la formule suivante avec la clé *(d,n) :* $m = c^d \bmod(n) = 1\ 570^{1\ 019} \bmod(3\ 337) = 688$.

Comme l'illustrent les formules précédentes, nous avons $m^{ed} = m \bmod(n)$.

Les explications mathématiques suivantes valident cette formule.

Comme e et d sont inverses modulo $(p - 1)(q - 1)$, $ed = 1 + k(p - 1)(q - 1)$ pour un certain entier k.

Dans le cas où m (le mot à chiffrer) $\neq 0 \bmod(p)$, nous avons :

$m^{ed} = m^{1 + k(p-1)(q-1)} \bmod(p)$

$m^{ed} = m(m^{(p-1)})^{k(q-1)} \bmod(p)$

D'après le théorème de Fermat, si p est premier, $a^{p-1} = 1 \bmod(p)$ pour tout $a \in Zp^*$.

Nous avons donc :

$m^{ed} = m(1)^{k(q-1)} \bmod(p)$

$m^{ed} = m \bmod(p)$

Par ailleurs :

$m^{ed} = m \bmod(p)$ si $m = 0 \bmod(p)$

Pour tout m, nous avons donc :

$m^{ed} = m \bmod(p)$

Nous pouvons montrer de la même manière que, pour tout *m :*

$m^{ed} = m \bmod(q)$

D'après un corollaire du théorème du reste chinois, si $n_1, n_2, \ldots, n_k$ sont premiers entre eux deux à deux et si $n = n_1 n_2 \ldots n_k$, pour deux entiers x et a quelconques, $x = a \bmod(n_i)$, pour $i = 1,2 \ldots k$, si et seulement si $x = a \bmod(n)$.

Nous avons donc pour tout *m,* si $n_1 = p$ et $n_2 = q$:

$m^{ed} = m \bmod(p \times q)$

$m^{ed} = m \bmod(n)$

La sécurité de RSA réside dans la difficulté de factoriser un grand nombre n (comme les clés publique et privée sont fondées sur p et q, un attaquant doit factoriser n pour casser

le chiffrement). En effet, déduire les facteurs premiers d'un grand nombre *n* est un problème difficile, que l'on ne sait pas résoudre efficacement (impossibilité calculatoire) pour des grands nombres.

À l'heure actuelle, il est donc impératif d'utiliser pour RSA des entiers *p* et *q* tels que leur produit comporte au moins 1 024 bits.

Extraction de logarithmes discrets

Outre la factorisation entière, un autre problème, largement répandu en cryptographie, concerne l'extraction de logarithmes discrets et s'exprime de la façon suivante : étant donné un groupe fini *G* noté multiplicativement, un générateur *g* de *G* et un élément *b* dans *G,* trouver *x* dans {0 ... |*G*|– 1} tel que $b = g^x$.

Ce problème est à l'origine du protocole d'échange de clés de Diffie-Hellman. Ces dernières années, son adaptation à d'autres groupes a donné lieu à ce qu'on a appelé les cryptosystèmes sur courbes elliptiques.

L'idée est d'utiliser comme groupe *G* le groupe additif des points d'une courbe elliptique sur un corps fini *F* (groupe additif $(EC(GF(2^m)),+)$. Sans entrer dans les détails, nous pouvons dire qu'il n'est pas connu d'algorithme sous-exponentiel qui résolve le problème du logarithme discret dans ce contexte, contrairement au problème du logarithme discret dans le groupe multiplicatif *G* d'un corps fini (groupe multiplicatif $(Z/pZ,*)$).

Cette dernière observation a pour conséquence importante de permettre l'utilisation de clés de taille moindre comparée à celles nécessaires aux cryptosystèmes fondés sur le logarithme discret dans les groupes classiques. À l'heure actuelle, 170 bits de clé (groupe additif $(EC(GF(2^m)),+)$ suffisent pour assurer le même niveau de sécurité qu'une clé RSA de 1 024 bits (groupe multiplicatif $(Z/pZ,*)$).

À titre d'exemple, la séquence d'échanges qui suit permet de chiffrer et de signer un message à l'aide de l'algorithme RSA :

1. Avec la paire de clés générée (privée/publique), John peut créer une signature électronique de son message certifiant que c'est bien lui qui a créé le message. Pour ce faire, John passe le message à transmettre dans une fonction de hachage afin de créer une empreinte unique du message.

2. John chiffre cette empreinte avec sa clé privée afin d'obtenir la signature électronique de John pour le message à transmettre. La clé privée de John étant unique et non diffusée, il est le seul à pouvoir obtenir cette signature *(voir figure 8.4)*.

3. Une fois le message signé, John envoie le message et la signature à Joe. John peut aussi chiffrer le message avec la clé publique de Joe afin d'assurer la confidentialité du message *(voir figure 8.5)*.

4. Pour vérifier la signature électronique de John, Joe fait passer le message reçu dans la même fonction de hachage que John.

5. En parallèle, il déchiffre la signature de John avec la clé publique de John.

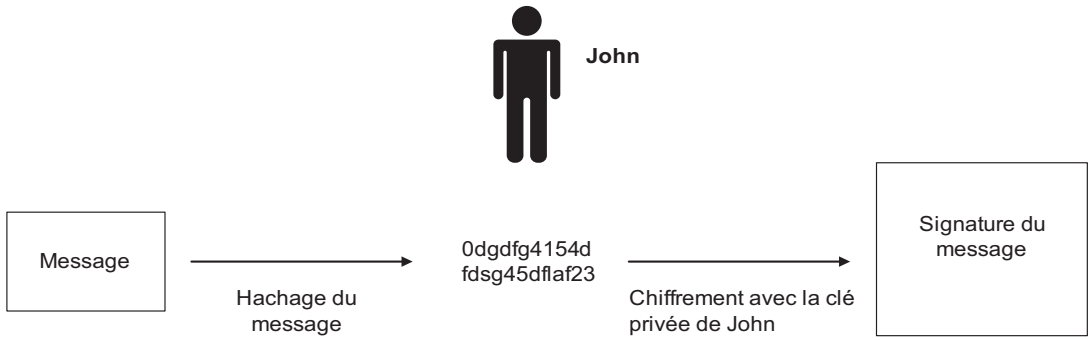


Figure 8.4

Signature d'un message par John

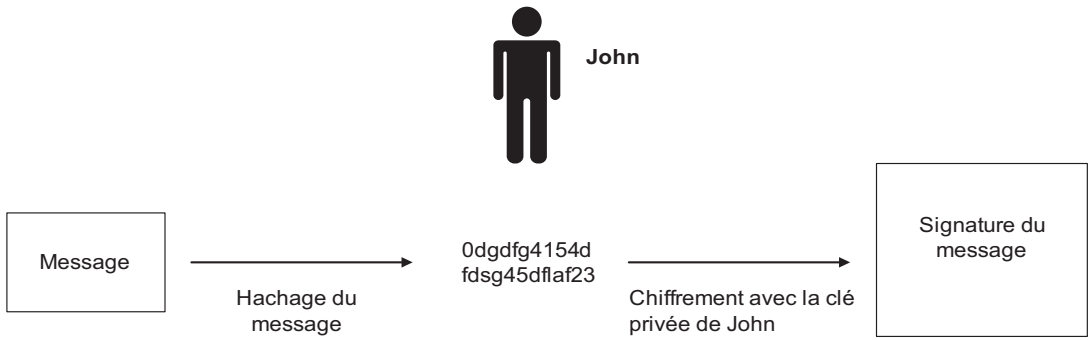


Figure 8.5

John envoie un message chiffré et signé à Joe

6. Les deux actions précédentes lui permettent d'obtenir deux empreintes, celle du message reçu et celle du message envoyé par John. Si les deux empreintes sont égales, c'est le message original écrit par John. Sinon, il y a problème *(voir figure 8.6)*.

La vérification d'une signature peut être réalisée à l'aide d'un programme informatique intégré, par exemple, à un logiciel de messagerie.

Certificats électroniques

Un certificat électronique est une assurance de sécurité sur l'identité électronique d'un individu ou d'un système. Les infrastructures PKI (Public Key Infrastructure) sont conçues pour mettre en œuvre l'architecture correspondante.

Une PKI est une infrastructure composée d'un ensemble de systèmes, de procédures et de politiques, dont les fonctions sont les suivantes :

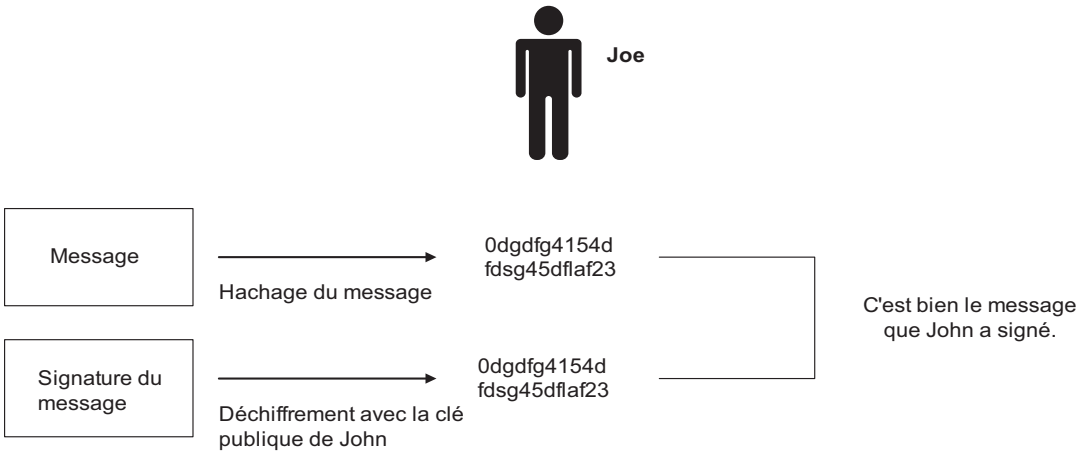


Figure 8.6
Joe vérifie le message envoyé par John

- enregistrer les entités désirant obtenir des certificats électroniques ;
- fabriquer des biclés, c'est-à-dire des paires de clés privée et publique ;
- certifier des clés publiques afin de créer des certificats et de publier ces derniers sur des annuaires publics, généralement des serveurs LDAP ;
- révocation de certificats et gestion de listes de révocation.

L'obtention d'un certificat numérique doit suivre des procédures et politiques très strictes, comme l'illustre la figure 8.7.

Les chiffres indiqués sur les flèches de la figure indiquent le séquencement des étapes pour délivrer un certificat électronique. De manière très simplifiée, l'utilisateur désirant obtenir un certificat électronique fait une demande auprès de l'autorité d'enregistrement (AE). Après validation de l'identité du demandeur, l'AE génère un couple de clés (publique, privée), envoie la clé privée suivant des procédures sécurisées à l'utilisateur (chemin de confiance) et certifie la clé publique par l'autorité de certification en apposant sa signature électronique sur le certificat. Le certificat est alors installé sur un annuaire public accessible à tous.

Un certificat électronique, ou passeport numérique, contient toutes les informations relatives à l'identité d'une personne, ainsi que d'autres champs non détaillés ci-dessous :

- numéro de version associé au certificat, par exemple X.509 v3 ;
- numéro de série fourni par l'autorité de certification ayant délivré le certificat ;
- algorithme utilisé pour la signature du certificat ;
- nom de l'autorité ayant délivré le certificat ;

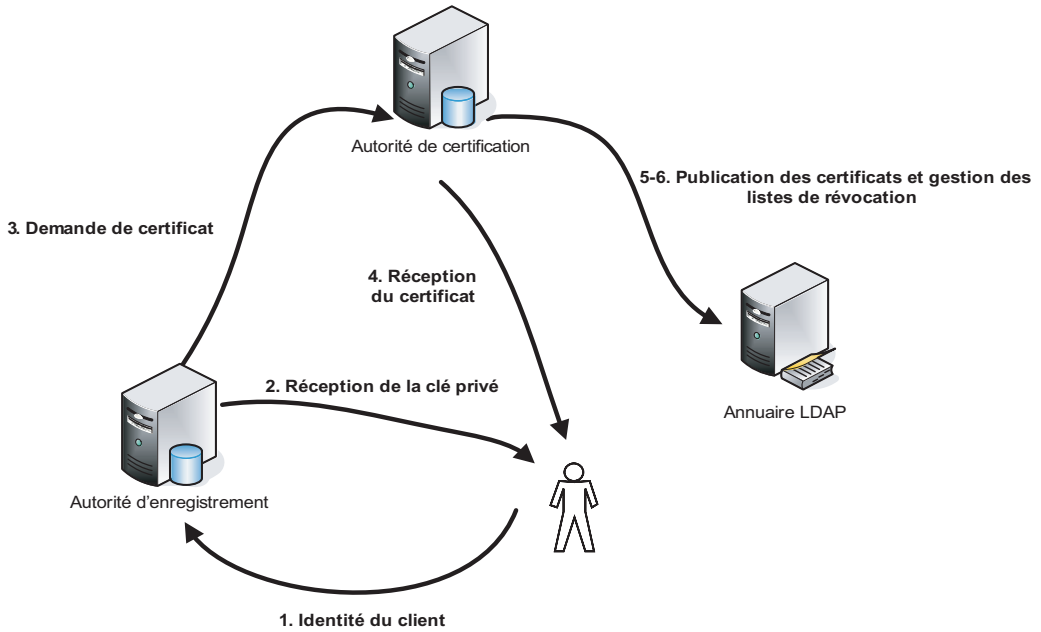


Figure 8.7
Les échanges dans une infrastructure à clé publique

- date de validité du certificat (dates de création et d'expiration) ;
- nom de la personne de destination du certificat ;
- clé publique de la personne certifiée.

D'autres informations concernant des attributs spécifiques associés au certificat dépendent de la version du certificat, etc.

À partir de ces informations, dont l'autorité de certification vérifie préalablement la validité, cette même autorité de certification génère une signature de certification en créant dans un premier temps une empreinte de ces informations grâce à un algorithme de hachage et en chiffrant cette empreinte par un algorithme de chiffrement asymétrique grâce à la clé privée de l'autorité de certification.

La figure 8.8 illustre le processus de création d'un certificat électronique par le hachage des informations concernant John puis par la création de la signature en chiffrant avec la clé privée de l'autorité de certification le résultat de la fonction de hachage.

Pour vérifier une signature d'une autorité de certification, il suffit de prendre l'ensemble des informations du certificat, excepté la signature, afin de créer une empreinte puis de déchiffrer la signature de l'autorité de certification grâce à la clé publique de cette même autorité afin de retrouver l'empreinte initiale certifiée. La dernière étape consiste à

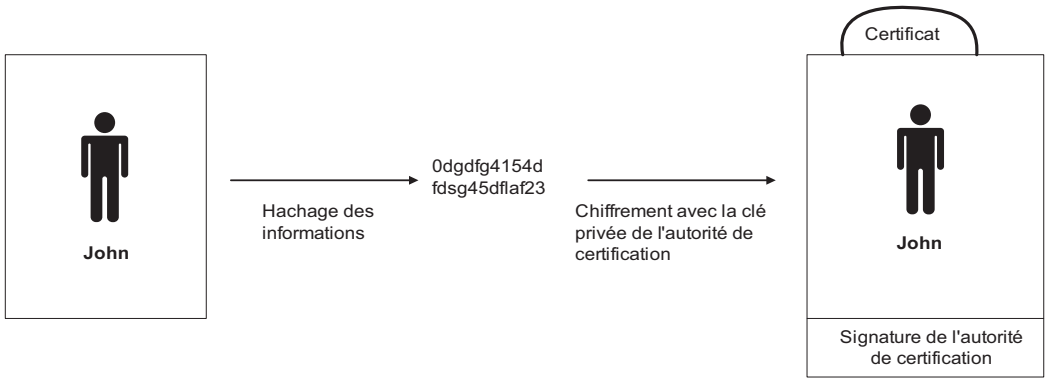


Figure 8.8
Signature de l'autorité de certification

comparer les deux empreintes. Si elles correspondent, le certificat est valide, sinon il ne peut être considéré comme de confiance.

La figure 8.9 illustre le processus de vérification de la validité d'un certificat électronique en comparant les empreintes du certificat à celle signée par l'autorité de certification.

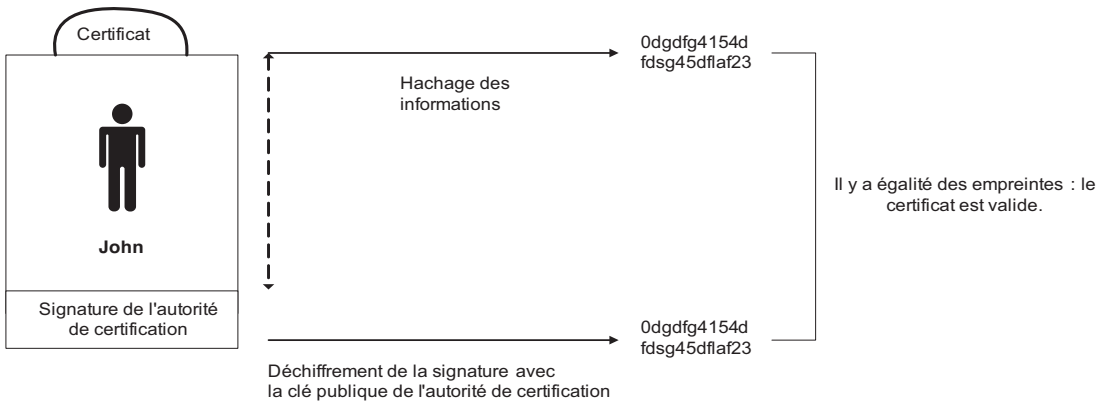


Figure 8.9
Vérification de la validité d'un certificat électronique

Un certificat est disponible dans le domaine public. En revanche, la clé privée associée au certificat est précieusement protégée sur un support physique sécurisé, tel qu'une carte à puce, ou token. L'accès à la carte à puce est protégé par un code PIN afin d'assurer une authentification forte de l'individu.

L'utilisation de biclés certifiées entraîne la publication en toute confiance de la clé publique. Cette publication doit assurer la validité de la clé et son appartenance à la bonne personne.

La publication des certificats des clés publiques utilise les structures d'annuaires de type LDAP (Lightweight Directory Access Protocol), définies dans la RFC 2251. Les certificats révoqués sont regroupés dans des listes de révocation, ou CRL (Certificate Revocation List). Les CRL sont des structures de données signées, dont le format est défini par le protocole X.509 v2. Ce format peut permettre une distribution des CRL *via* les annuaires LDAP tels que Netscape Directory Server d'iPlanet.

L'implémentation d'une PKI est un projet essentiellement organisationnel, dont la dimension technique représente moins de 10 % des efforts (configuration des plates-formes et du réseau, implémentation du produit PKI, etc.). Les 90 % restants concernent les aspects organisationnels, tels que la conception de la stratégie de sécurité, la constitution de l'annuaire définissant le choix du référentiel d'entreprise (gestion des prestataires et stagiaires, règles de nommage, etc.), l'identification des responsabilités, l'élaboration de la politique de certification et la rédaction de la déclaration des pratiques de certification.

Comme expliqué précédemment, les PKI offrent une assurance de sécurité pour un certificat électronique. Ce dernier peut être utilisé avec des applications telles que l'e-mail chiffré, le réseau privé virtuel fondé sur IPsec, le commerce électronique, etc.

Comme les PKI intègrent la cryptographie à clé publique et les certificats électroniques, elles peuvent être confiées à des tiers de confiance, lesquels doivent en retour recevoir l'agrément de la DCSSI (Direction centrale de la sécurité des systèmes d'information) pour avoir une portée nationale. Cependant, l'absence de standards pour l'implantation des PKI pose de sérieux problèmes d'interopérabilité entre les différentes offres du marché.

Paires de clés PGP (Pretty Good Privacy)

Conçu par Phil Zimmermann, PGP a pour fonction d'offrir des services de confidentialité et d'authentification pour la messagerie électronique et le stockage de données.

Le succès planétaire de ce logiciel vient notamment de ce qu'il est gratuit (pour un usage personnel et non commercial) et disponible sur la plupart des systèmes d'exploitation actuels. La certification, ou plus précisément les niveaux de confiance définis des clés privée/publique créées, est indépendante des organismes de standardisation, contrairement à PKI.

PGP se fonde sur les algorithmes considérés comme les plus sûrs actuellement et largement diffusés. Citons notamment les algorithmes de chiffrement à clé publique RSA, DSS, Diffie-Hellman, etc., les algorithmes à clé partagée IDEA, CAST-1, etc., et les fonctions de hachage SHA-1, etc. Des fonctions de compression sont également disponibles afin de limiter la taille des messages transitant sur le réseau.

Chaque utilisateur possède une ou plusieurs paires de clés privée/publique et diffuse les parties publiques de ces clés aux personnes avec lesquelles il désire communiquer. Comme expliqué précédemment, un utilisateur utilise sa paire de clés privée/publique afin de s'authentifier et de chiffrer et signer ses messages.

Voici un exemple de biclé (privée/publique) générée par PGP :

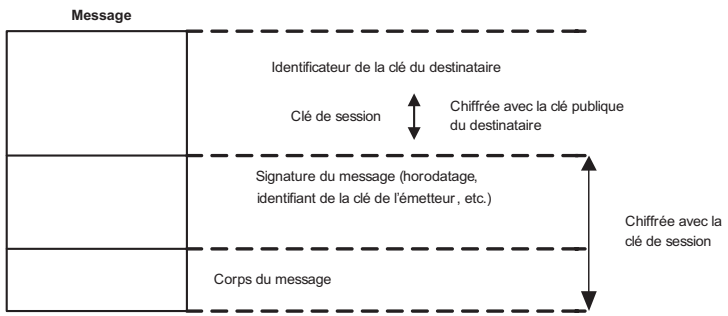
```
-----BEGIN PGP PRIVATE KEY BLOCK-----
Version: PGP 6.5.1fr pour usage non commercial

lQHPBD5skkORBADhHkEAqKlQ8DerYhmlC3XYObqFt8N/mZmOa7/b+3sky81q+7E4
Y59+JP59snciOiG1xgFTE++m4VV9+dJbIoWT/OQkOhVP/zyaAZyKJIeiO/+Ui7td
Nu2zcu4iKBGFdwRVuPrOReZakOwLitWmKdDeEziyqsxeNH1BH7EWqLT8+QCg/xCB
5GZNyijc91KJ98owF1PtAc8EANXTpIOt/Kzw/7CkHiZ1fMN31Po5YAFwlM25erHL
macsDAe81OK4HO9g9YTUtzXsXjrurgduFhao7L8RqoB+Vcp9AiCJOABbDGPKL87e
9yePLwl9EcnCyI/6kcLdkGU5A64F+O8UwoU7Hjgkz6pQxOptv6RR6X3v7IOuGzVB
+lHoBADBrOtrvB2bIwRGc8wvY9dDU/dxvOZo6BdCXYVeaV1nLeOSxGHZGi1p84xd
OtzgafyPH4fzK5baJoFJsJAnC8Onij3G5o3DkfwPk7v+TxvdPD3edls9Exx8Gv8C
VCQc5KItgvTn7JnDmADriYKfb2n6c4UtFxEsCHTGaxOjAyLdWv8DAwLssclIzhgg
XWB3Xx/+c+ApZ2Y7jOcTaFoTe24++vUrIeJCdI1aHR2VW7QqY2VkcmljIGxsb3Jl
bnMgPGNlZHJpYy5sbG9yZW5zQGVxdWFudC5jb2O+nQJRBD5skkOQCAD2Qle3CH8I
F3KiutapQvMF6PlTETlPtvFuuUs4INoBp1ajFOmPQFXzOAfGyOOplK33TGSGSfgM
g71l6RfUodNQ+PVZX9x2Uk89PY3bzpnhV5JZzf24rnRPxfx2vIPFRzBhznzJZv8V
+bv9kV7HAarTW56NoKVyOtQa8L9GAFgr5fSI/VhOSdvNILSd5JEHNmszbDgNRROP
fIizHHxbLY7288kjwEPwpVsYjY67VYy4XTjTNP18F1dDoxOYbN4zISy1Kv884bEp
QBgRjXyEpwpy1obEAxnIByl6ypUM2Zafq9AKUJsCRtMIPWakXUGfnHy9iUsiGSa6
q6Jew1XpMgs7AAICB/4nZOhHEjdjo9hRdwhCmHYxYjm+iq14iCJil/WHyzhpkqQN
7QQyFPMNNtuw1Dy7qxQ31IEPiyRf1jS4atVbP1Fl+63g4E+Kk9lSchkZmaLv1fPV
xY+Mci8FpQ1R8w7jN/Bxwn1l1yxryNbVphDhuLP8ehruGvmRrWuK7KpJS/UDJIHT
S4JxOlPM+GgIW6l4+lQzy7ImKQdEhfqGfG/vyOnQNUva4Ww4r3Q+4fhZECMpQzgZ
IFZ5ujLSuNbUDakPHAYJS3OSXwVyUhQhDslOhURXpJeB292VeRh3rFhIOS4v6W5E
5aYATIM/9xac7IOg5Z9lQBPr3Lat+6WN32K/QwoN/wMDArAZzOZ1Q/DhYK9nsSfZ
xTChMFcHa175bjuqMya3AiECJOz1V3SorBIjpEnBAbAfVbNDJBu9pWlCS88fdO1H
QDM=
=EKBV
-----END PGP PRIVATE KEY BLOCK-----

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: PGP 6.5.1fr pour usage non commercial

mQGiBD5skkORBADhHkEAqKlQ8DerYhmlC3XYObqFt8N/mZmOa7/b+3sky81q+7E4
Y59+JP59snciOiG1xgFTE++m4VV9+dJbIoWT/OQkOhVP/zyaAZyKJIeiO/+Ui7td
Nu2zcu4iKBGFdwRVuPrOReZakOwLitWmKdDeEziyqsxeNH1BH7EWqLT8+QCg/xCB
5GZNyijc91KJ98owF1PtAc8EANXTpIOt/Kzw/7CkHiZ1fMN31Po5YAFwlM25erHL
macsDAe81OK4HO9g9YTUtzXsXjrurgduFhao7L8RqoB+Vcp9AiCJOABbDGPKL87e
9yePLwl9EcnCyI/6kcLdkGU5A64F+O8UwoU7Hjgkz6pQxOptv6RR6X3v7IOuGzVB
+lHoBADBrOtrvB2bIwRGc8wvY9dDU/dxvOZo6BdCXYVeaV1nLeOSxGHZGi1p84xd
OtzgafyPH4fzK5baJoFJsJAnC8Onij3G5o3DkfwPk7v+TxvdPD3edls9Exx8Gv8C
VCQc5KItgvTn7JnDmADriYKfb2n6c4UtFxEsCHTGaxOjAyLdWrQqY2VkcmljIGxs
b3JlbnMgPGNlZHJpYy5sbG9yZW5zQGVxdWFudC5jb2O+iQBOBBARAgAOBQI+bJJN
BAsDAgECGQEACgkQjMO/1DlHtZ8OqgCeI1nY7/b3eo7rfCFgcOfQhONw+RIAoOrr
iJ65E8egvMGFnOAvxmM1H5fGuQINBD5skkOQCAD2Qle3CH8IF3KiutapQvMF6PlT
ETlPtvFuuUs4INoBp1ajFOmPQFXzOAfGyOOplK33TGSGSfgMg71l6RfUodNQ+PVZ
X9x2Uk89PY3bzpnhV5JZzf24rnRPxfx2vIPFRzBhznzJZv8V+bv9kV7HAarTW56N
oKVyOtQa8L9GAFgr5fSI/VhOSdvNILSd5JEHNmszbDgNRROPfIizHHxbLY7288kj
wEPwpVsYjY67VYy4XTjTNP18F1dDoxOYbN4zISy1Kv884bEpQBgRjXyEpwpy1obE
```

```
AxnIByl6ypUM2Zafq9AKUJsCRtMIPWakXUGfnHy9iUsiGSa6q6Jew1XpMgs7AAIC
B/4nZOhHEjdjo9hRdwhCmHYxYjm+iq14iCJil/WHyzhpkqQN7QQyFPMNNtuw1Dy7
qxQ31IEPiyRf1jS4atVbP1Fl+63g4E+Kk9lSchkZmaLv1fPVxY+Mci8FpQ1R8w7j
N/Bxwn1l1yxryNbVphDhuLP8ehruGvmRrWuK7KpJS/UDJIHTS4JxOlPM+GgIW6l4
+lQzy7ImKQdEhfqGfG/vyOnQNUva4Ww4r3Q+4fhZECMpQzgZIFZ5ujLSuNbUDakP
HAYJS3OSXwVyUhQhDslOhURXpJeB292VeRh3rFhIOS4v6W5E5aYATIM/9xac7IOg
5Z9lQBPr3Lat+6WN32K/QwoNiQBGBBgRAgAGBQI+bJJNAAoJEIzDv9Q5R7WfffgA
oOfg7MnAs59Txk8RD/drg29aJevZAKDeXagEkodYGbiEGTN/86yPkdIXrQ==
=Yq7H
-----END PGP PUBLIC KEY BLOCK-----
```

Le chiffrement du message est réalisé à l'aide d'un algorithme de chiffrement symétrique, dont la clé est elle-même chiffrée par la clé publique de l'interlocuteur et ajoutée au message chiffré à transmettre. De la sorte, seul l'interlocuteur peut déchiffrer la clé de chiffrement symétrique avec sa clé privée et déchiffrer dans un second temps le message avec cette même clé de chiffrement symétrique.

La figure 8.10 illustre le format d'un message PGP.

Figure 8.10
Format d'un message PGP



Les paires de clés d'un utilisateur sont stockées localement dans des anneaux de clés. Un anneau de clés privées et un anneau de clés publiques contiennent l'ensemble des informations relatives aux clés. Les clés privées sont chiffrées par le biais d'une phrase (et non d'un mot) de passe grâce à un algorithme de chiffrement symétrique dont la clé est déduite par la phrase de passe.

Un anneau de clés privées contient les champs horodatage (date et heure à laquelle la clé a été produite), identifiant de clé (assurant que la clé est unique pour un utilisateur donné), clé publique, clé privée et identifiant utilisateur (généralement un e-mail).

Un anneau de clé publique contient les champs horodatage (date et heure à laquelle la clé a été produite), identifiant de clé (assurant que la clé est unique pour un utilisateur donné), clé publique, propriétaire de confiance (nous détaillons ce champ dans la suite du chapitre), identifiant utilisateur (généralement un e-mail), champ légitimité de clé (nous détaillons ce champ dans la suite du chapitre), signature et signature de confiance (plusieurs signatures peuvent être associées à une clé, certifiant par ce biais le degré de confiance de la clé).

La caractéristique principale de PGP est qu'il ne s'appuie pas sur des autorités de certification pour attribuer un niveau de confiance à une paire de clé donnée.

La définition de la confiance est caractérisée comme une relation :

- **Binaire.** J'ai ou je n'ai pas confiance.
- **Non symétrique.** Ce n'est pas parce que Cédric fait confiance à Laurent que Laurent fait confiance à Cédric.
- **Non transitive.** Ce n'est pas parce que Cédric fait confiance à Denis et que Denis fait confiance à Laurent, que Cédric fait confiance à Laurent.

Sachant qu'un certificat est finalement une assurance de sécurité sur la confiance que l'on peut porter à une clé publique, l'originalité de PGP est de traiter cette confiance sans autorité centrale, de la même manière que nous portons notre confiance à des individus.

Rappelons que la gestion de la confiance a pour objet de détecter de possibles fausses paires de clés, d'assurer par un degré de confiance l'appartenance d'une paire de clés à un individu donné et de garantir que tout utilisateur puisse signer une clé publique donnée en se fondant sur ce degré de confiance.

PGP associe à chaque clé publique les trois champs de confiance suivants :

- **Confiance de propriétaire.** Indique le degré de confiance mis dans une clé publique donnée. Cette valeur est directement renseignée par l'utilisateur.
- **Confiance de signature.** Indique le degré de confiance que l'utilisateur accorde à chaque signature associée à une clé publique donnée. Cette valeur est directement calculée par PGP en vérifiant dans l'anneau des clés publiques de l'utilisateur le degré de confiance de propriétaire de l'auteur de la signature.
- **Légitimité de clé.** Indique le degré de confiance que PGP peut accorder à la validité de l'appartenance d'une clé publique par rapport à un utilisateur donné. Cette valeur est directement calculée par PGP en se fondant sur l'ensemble des champs Confiance de signature associés à une clé publique.

À partir de ces champs, un utilisateur donné peut signer en toute confiance, ou certifier, une clé publique. Ce modèle est en tout point semblable au comportement que l'on peut adopter afin d'établir une relation de confiance avec autrui.

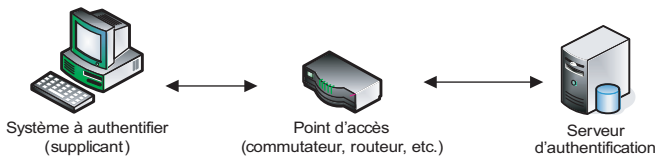
La révocation d'une clé publique est évidemment autorisée. Il suffit que le propriétaire émette un certificat de révocation de la clé publique et le diffuse le plus rapidement possible à tous ses correspondants de façon que chacun puisse mettre à jour ses bases de clés.

Assurer le contrôle des accès physiques à un réseau local

Le protocole IEEE 802.1X est un standard dont l'objectif est de fournir un mécanisme d'autorisation de l'accès physique à un réseau local après authentification (le réseau peut être filaire ou sans fil).

Les composants qui interviennent dans un tel mécanisme son le système à authentifier (supplicant), le point d'accès au réseau local (commutateur, routeur, etc.) et le serveur d'authentification *(voir figure 8.11)*.

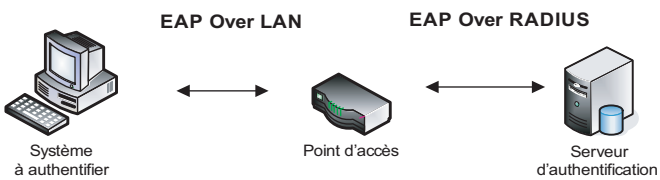
Figure 8.11
Composants de l'accès au réseau local



Tant que le système n'est pas authentifié, il ne peut avoir accès au réseau local hormis les échanges entre le système et le serveur d'authentification.

Pour un réseau filaire, le dialogue entre le système à authentifier et le point d'accès se fonde sur le protocole EAP (Extensible Authentication Protocol) pour réaliser l'authentification du système (EAP Over LAN). En revanche, le point d'accès et le serveur d'authentification dialoguent à l'aide du protocole EAP Over RADIUS (Remote Authentication Dial-In User Service), comme l'illustre la figure 8.12.

Figure 8.12
Protocoles d'accès au réseau local

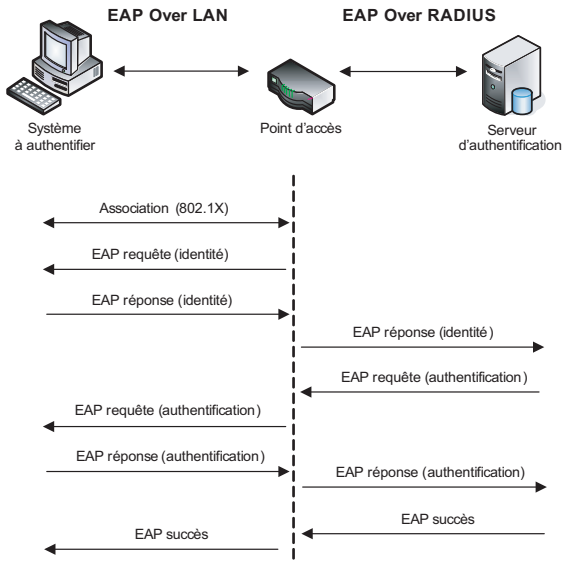


Les messages EAP peuvent être de quatre types : requêtes, réponses, succès et échec. La figure 8.13 illustre une authentification réussie.

Le protocole 802.1X ne propose pas une seule méthode d'authentification mais repose sur les différentes possibilités d'authentification véhiculées par le protocole EAP, notamment les suivantes :

- EAP-MD5 : pas d'authentification mutuelle ; le client s'authentifie à l'aide d'un mot de passe.
- LEAP : protocole propriétaire de Cisco s'appuyant sur une authentification de type challenge/réponse, dérivée de la méthode MS-CHAP de Microsoft, fondée sur un couple identifiant/mot de passe**.**
- EAP-TLS (Transport Layer Security) : authentification mutuelle entre le client et un serveur fondée sur des certificats.
- EAP-TTLS (Tunneled Transport Layer Security) ou EAP-PEAP (Protected EAP) : authentification mutuelle entre le client et un serveur fondée sur un certificat côté serveur et pouvant être réalisée par un couple compte/mot de passe côté client. Dans ce dernier cas, un tunnel TLS s'établit avant que le client ne transmette ses éléments d'authentification à partir d'un couple identifiant/mot de passe.

Figure 8.13
Accès au réseau local et authentification



- EAP-FAST (Flexible Authentication via Secure Tunneling) : protocole développé par Cisco et disponible depuis avril 2004, qui utilise le chiffrement à clé symétrique entre le serveur et le client pour créer un tunnel TLS lors de l'échange des données d'authentification de la part du client.

Après une authentification positive et suivant le type d'équipement associé au point d'accès, il est possible d'appliquer des politiques de sécurité, telles que l'affectation de l'accès au système dans un VLAN dédié (Virtual LAN), des règles de filtrage spécifiques, etc.

Assurer le contrôle des accès distants classiques

Les accès distants au réseau d'entreprise traversent généralement des réseaux publics. Ces derniers offrent la capillarité nécessaire pour garantir des connexions à des coûts locaux.

Ces réseaux publics sont soit le réseau téléphonique de bout en bout, soit le réseau téléphonique pour l'accès puis le réseau Internet jusqu'au réseau d'entreprise, soit encore des accès *x*DSL, Numéris, etc., à Internet ou à des réseaux fermés d'opérateurs de télécommunications jusqu'au réseau d'entreprise. Nous appelons réseaux fermés des réseaux desservant des protocoles de type X.25, qui offrent un cloisonnement des trafics réseau.

Il est primordial de protéger les PC portables des utilisateurs d'accès distants contre les pénétrations directes, par virus, etc., pouvant entraîner le vol de mots de passe ou d'autres moyens d'authentification non suffisamment protégés. Malgré tous les dispositifs mis en place, l'accès est alors rapidement usurpé.

Les moyens de protection des ordinateurs portables doivent s'appuyer à la fois sur un pare-feu local implémentant des règles très restrictives et un système antivirus régulière-

ment mis à jour. Ces éléments doivent être sous le contrôle exclusif d'un groupe d'administrateur afin d'éviter toute erreur de configuration d'un utilisateur.

Règles de sécurité pour le contrôle des accès distants

Les règles de sécurité à considérer pour assurer le contrôle des accès distants sont les suivantes :

- Les accès distants sont authentifiés et chiffrés pour toute connexion au réseau d'entreprise.
- Les adresses IP associées à des accès distants sont situées dans une classe d'adresses IP bien déterminée afin de faciliter les filtrages ultérieurs par d'autres équipements de sécurité.
- Les services offerts pour les accès distants sont limités aux besoins identifiés. Aucun accès à une information sensible n'est autorisé pour les accès distants.
- Les ordinateurs utilisés pour les accès distants implémentent un logiciel antivirus ainsi qu'un pare-feu local. La configuration est établie à l'avance et correspond aux standards de sécurité de l'entreprise.
- Des contrôles de sécurité réguliers des accès distants sont menés à la fois sur les serveurs hébergeant les logiciels d'accès distants et sur les bases de données où sont définis et autorisés les utilisateurs.
- La base de données des utilisateurs autorisés à accéder à distance est périodiquement auditée afin d'éliminer les comptes non utilisés.

Le choix du niveau de tunneling et de sécurité à mettre en œuvre dépend de la maîtrise que l'on a du réseau. Par exemple, une entreprise devrait fonder sa sécurité sur des tunnels de niveau 3 plutôt que sur des tunnels de niveau 2 du fait qu'elle ne maîtrise pas les artères de connexion.

Le tableau 8.1 compare les caractéristiques des différents protocoles d'accès distants détaillés dans les sections qui suivent.

Tableau 8.1 Caractéristiques des protocoles d'accès distants

	L2TP	PPTP	IPsec
Mode	Client-serveur, tunnel opérateur (L2F)	Client-serveur	Client-client, tunnel passerelle
Utilisation	Accès distant *via* un tunnel	Accès distant *via* un tunnel	Intranet, extranet, accès distant
Protocole transporté	IP, IPX, NetBEUI, etc.	IP, IPX, NetBEUI, etc.	IP
Service de tunnel	Point-à-point	Point-à-point	Multipoint
Niveau OSI	2 (encapsulé dans IP)	2 (encapsulé dans IP)	3
Partage du tunnel	Oui	Oui	Oui
Authentification utilisateur	PAP, CHAP, EAP, SPAP	PAP, CHAP, EAP, SPAP	Non
Authentification du paquet	Le tunnel peut être authentifié.	Le tunnel peut être authentifié.	Oui, *via* l'en-tête AH
Chiffrement du paquet	Oui, *via* un tunnel IPsec	Oui, *via* la couche MPPE spécifique de Microsoft	Oui, *via* l'en-tête ESP
Affectation d'adresses dynamique	Oui (PPP NCP)	Oui (PPP NCP)	Selon les implémentations
Gestion des clés	Non	Non	IKE, SKIP
Résistance aux attaques	Non	Non	Oui

PPP (Point-to-Point Protocol)

Les protocoles associés aux accès distants sont des protocoles de type point-à-point et tunnel, capables de faire transiter sur les réseaux des trafics de sessions qui tiennent compte des contraintes multiprotocolaires. Le transport simultané de protocoles différents, tels que IP, IPX ou NetBEUI (NetBios Extended User Interface), peut être encapsulé par le protocole PPP (Point-to-Point Protocol).

Dans une connexion à distance, le protocole entre l'ordinateur portable, incluant le modem, et le réseau d'interconnexion, est généralement PPP (Point-to-Point Protocol). Ce protocole d'encapsulation de paquets est lui-même composé de sous-protocoles chargés du contrôle de liaison, ou LCP (Link Control Protocol), et du contrôle réseau, ou NCP (Network Control Protocol).

Le contrôle réseau NCP comporte les sous-protocoles suivants :

- ATCP-AppleTalk
- BCP-Bridging
- BVCP-Banyan Vines
- CCP-PKZIP, Microsoft Point-To-Point Compression, etc. (avec compression)
- DNCP-DECnet Phase IV
- ECP-DES, triple-DES, etc. (avec chiffrement)
- IPCP-Internet
- IPv6CP-IPv6
- IPXCP-IPX
- NBFCP-NetBIOS
- OSINLCP (couches réseau OSI)
- PPP-LEX-LAN
- SDCP-Serial Data
- SNACP-SNA
- XNSCP-XNS IDP

Le contrôle de liaison LCP comporte les sous-protocoles suivants :

- BACP (allocation de bande passante)
- LCP
- LQR (qualité des connexions)
- PAP (Password Authentication Protocol)
- CHAP (Challenge Handshake Authentication Protocol)
- EAP (Extensible Authentication Protocol)

Moyens d'authentification du protocole PPP

Plusieurs méthodes d'authentification sont disponibles avec le protocole PPP. Ces méthodes peuvent offrir une authentification élémentaire à l'aide de mots de passe jusqu'à une authentification fondée sur des certificats électroniques.

PAP (Password Authentication Protocol) est un protocole d'authentification qui utilise des mots de passe en texte clair. C'est le protocole d'authentification le plus faible.

MS-CHAP est un processus d'authentification mutuelle qui repose sur un cryptage unidirectionnel du mot de passe. Les étapes de ce processus sont les suivantes :

1. Le client fait une demande de connexion au serveur d'authentification d'accès distant.
2. Le serveur d'accès distant envoie une demande de vérification au client, qui consiste en un identificateur de session et une chaîne d'interrogation arbitraire.
3. Le client d'accès distant envoie une réponse contenant le nom de l'utilisateur et un cryptage unidirectionnel de la chaîne d'interrogation reçue contenant le mot de passe de l'utilisateur.
4. Le serveur d'authentification vérifie la réponse du client en appliquant le même cryptage unidirectionnel puisqu'il connaît le mot de passe de l'utilisateur contenu dans sa base de données. Il renvoie une réponse contenant l'indication du succès ou de l'échec de la tentative de connexion et une réponse authentifiée fondée sur la chaîne d'interrogation envoyée.
5. Le client d'accès distant vérifie la réponse d'authentification et, si celle-ci est correcte, utilise la connexion. Si la réponse d'authentification est incorrecte, le client d'accès distant interrompt la connexion.

Développé par Microsoft, MS-CHAP est fondé sur le protocole CHAP et sur des mots de passe préalablement cryptés de manière unidirectionnelle. Les bases de données d'authentification ne contiennent pas les mots de passe en clair.

Plutôt que de définir d'autres protocoles d'authentification, l'IETF a préféré définir un cadre générique indépendant de la méthode d'authentification. Le protocole EAP (Extensible Authentication Protocol) offre ce cadre générique et permet de transporter des données d'authentification entre un client et un serveur (RFC 3748). Il est ainsi possible de changer de méthode d'authentification sans changer le protocole EAP.

EAP est donc uniquement un protocole d'encapsulation, qui est principalement utilisé dans les environnements PPP et IEEE 802.11. Il ne comprend que quatre types de messages (requête, réponse, succès et échec), mais plusieurs dizaines de méthodes d'authentification sont disponibles, notamment les suivantes : MD5-Challenge, OTP (One Time Password), GTC (Generic Token Card), RSA Public Key Authentication, DSS Unilateral, KEA, KEA-VALIDATE, EAP-TLS, Defender Token (AXENT), RSA Security SecurID EAP, Arcot Systems EAP, EAP-Cisco Wireless, EAP-SIM, SRP-SHA1 Part 1, SRP-SHA1 Part 2, EAP-TTLS, Remote Access Service, EAP-AKA, EAP-3Com, PEAP, MS-EAP-Authentication, Mutual Authentication w/Key Exchange, CRYPTOCard, EAP-MSCHAP-V2, DynamID, Rob EAP, SecurID EAP, MS-Authentication-TLV, SentriNET,

EAP-Actiontec Wireless, Cogent Systems Biometrics Authentication EAP, AirFortress EAP, EAP-http, Digest, SecureSuite EAP, DeviceConnect EAP, EAP-SPEKE, EAP-MOBAC, EAP-FAST, EAP Flexible Authentication via Secure Tunneling, ZLXEAP, ZoneLabs EAP, EAP-Link, EAP-PAX, etc.

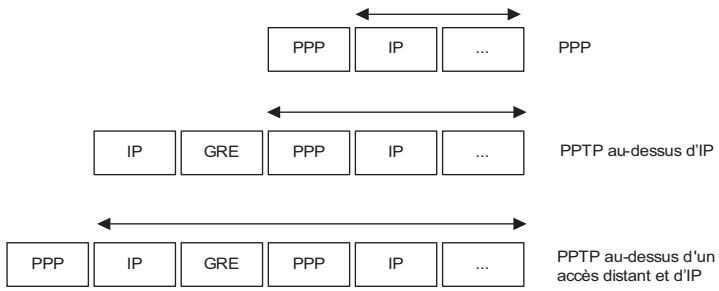
Le protocole EAP est conçu pour répondre à la demande croissante d'authentification des utilisateurs d'accès distants en employant d'autres périphériques de sécurité que les mots de passe. Grâce à ce protocole, il est possible d'ajouter la prise en charge de plusieurs modèles d'authentification, notamment les cartes à jeton, les mots de passe à usage unique, l'authentification par clé publique utilisant des cartes à puce, etc. Cela permet d'employer un serveur d'arrière-plan pour implémenter les divers mécanismes d'authentification, le serveur authentifiant se chargeant simplement de transmettre les éléments d'authentification.

PPTP (Point-to-Point Tunneling Protocol)

Le protocole PPTP permet de créer un réseau privé virtuel par la prise en charge de protocoles tels que IP, NetBEUI, IPX, etc. Ce protocole a été développé par Microsoft en collaboration avec Ascend et 3Com.

PPTP encapsule, par le biais d'un tunnel, les protocoles IP, IPX et NetBEUI, eux-mêmes encapsulés dans des paquets PPP. Il utilise pour cela le protocole GRE (Generic Routing Encapsulation), comme l'illustre la figure 8.14.

Figure 8.14
Encapsulation des trames PPP dans GRE



MPPE (Microsoft Point-to-Point Encryption) crypte les données des connexions d'accès distants PPP ou des connexions VPN PPTP. Les méthodes de chiffrement MPPE utilisent des clés de longueur variable, de 40 à 128 bits. Ces méthodes sont prises en charge par le chiffrement des données (RC4). MPPE assure la sécurité des données entre la connexion du client distant (connexion PPTP) et le serveur d'accès distant.

Les méthodes d'authentification de PPTP héritent des méthodes d'authentification du protocole PPP.

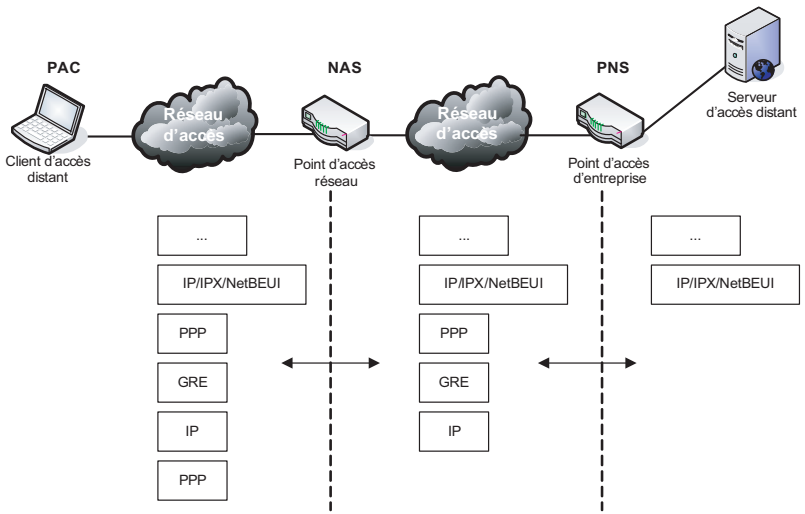
Pour établir une session PPTP, l'ordinateur client, ou PAC (PPTP Access Concentrator), se connecte à distance *via* le protocole PPP à un concentrateur d'accès NAS (Network Access Server) de son FAI. Puis il établit une seconde session au serveur réseau PPTP, ou

PNS (PPTP Network Server), afin de négocier les termes du tunnel PPTP. L'authentification de l'utilisateur est alors demandée afin de valider la session entrante en s'appuyant sur les méthodes d'authentification héritées de PPP.

Le tunnel établi sur le réseau IP consiste en une encapsulation de niveau 3 par le protocole IP/GRE des paquets PPP, comme illustré à la figure 8.15.

Figure 8.15

Couches réseau mises en œuvre dans l'accès distant PPTP



PPTP utilise en parallèle une connexion de contrôle entre le couple PAC-PNS *via* une session TCP sur le port 1723, de façon à transmettre les informations de contrôle et de gestion des appels PPTP, ainsi qu'un tunnel IP entre le couple PAC-PNS pour le transport des paquets PPP encapsulés par GRE (service numéro 47).

L2TP (Layer 2 Tunneling Protocol)

L2TP est un protocole de tunneling identique à bien des égards à PPTP et fondé sur la convergence des protocoles PPTP et L2F.

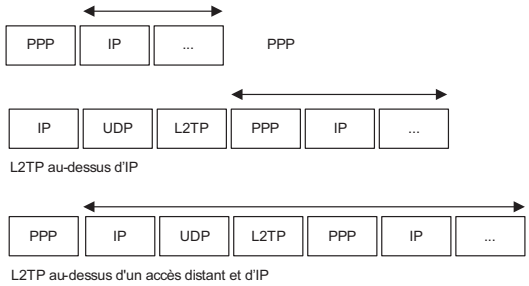
L2TP encapsule, par le biais d'un tunnel, les protocoles IP, IPX et NetBEUI, eux-mêmes encapsulés dans des paquets PPP. Il utilise pour cela des paquets IP/UDP sur les réseaux IP pour le transport des tunnels L2TP, comme illustré à la figure 8.16.

Contrairement à PPTP, L2TP n'utilise pas MPPE pour crypter les paquets PPP mais s'appuie sur les services de sécurité IPsec.

L'encapsulation des paquets L2TP dans IPsec consiste en une première encapsulation de la trame PPP (contenant un paquet IP ou IPX ou une trame NetBEUI) dans un en-tête UDP, suivie d'une encapsulation dans une trame IPsec.

Les méthodes d'authentification de L2TP héritent des méthodes d'authentification du protocole PPP.

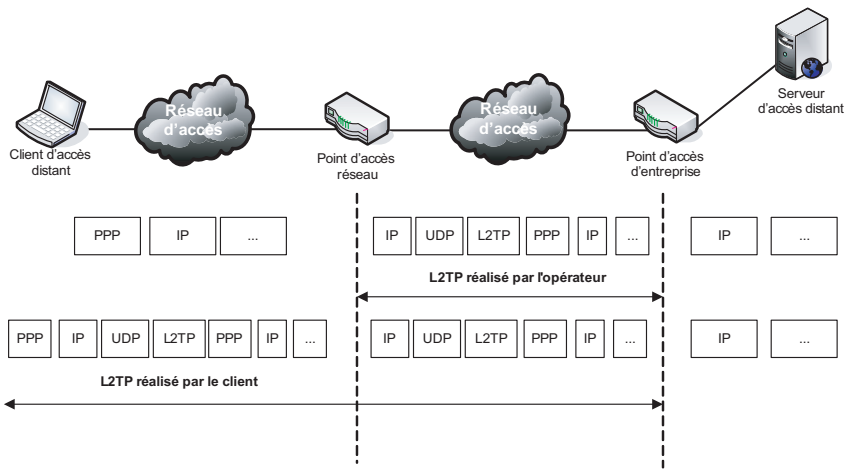
Figure 8.16
Encapsulation L2TP des trames PPP



Pour établir une session L2TP, le client se connecte à distance *via* le protocole PPP à un concentrateur d'accès L2TP, ou LAC (L2TP Access Concentrator), de son FAI. Ce dernier établit un tunnel vers le serveur réseau L2TP, ou LNS (L2TP Network Server), qui est généralement réalisé par un routeur. Il est aussi possible que la fonction de LAC soit directement réalisée par l'ordinateur client, comme nous le verrons par la suite.

L'authentification de l'utilisateur est demandée afin de valider la session entrante en s'appuyant sur les méthodes d'authentification héritées de PPP. Le tunnel établi sur le réseau IP consiste en une encapsulation de niveau 3 par le protocole IP/UDP des paquets PPP, comme illustré à la figure 8.17.

Figure 8.17
Couches réseau mises en œuvre pour un accès distant L2TP



L2TP utilise en parallèle, sur un tunnel donné entre le couple LAC-PNS, les messages de contrôle, de façon à gérer les sessions, ainsi que les paquets PPP encapsulés par L2TP et reposant sur UDP (port 1701). Dans le cas où le client gère la fonction L2TP, il doit gérer deux sessions PPP, l'une avec le point d'accès réseau et l'autre avec le point d'accès.

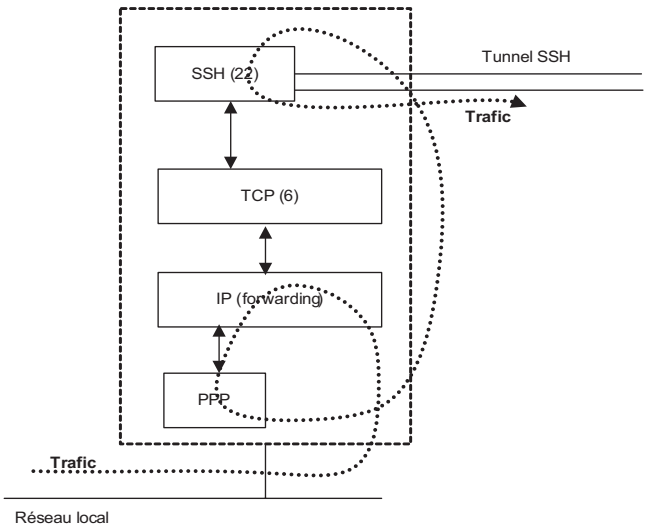
SSH (Secure SHell)

SSH permet de rediriger n'importe quel flux TCP en mode tunnel dans une session SSH. Le flux de l'application considérée est alors encapsulé à l'intérieur du tunnel créé par la connexion, ou session, SSH.

Le protocole PPP, qui est généralement utilisé pour établir une interconnexion à distance à un réseau en se positionnant au niveau de la couche 2 OSI, permet aussi, s'il est redirigé dans une session SSH, de créer un tunnel IP entre deux systèmes reliés par un réseau.

Il est ainsi possible de créer un réel tunnel IP à travers SSH en encapsulant tout d'abord le trafic IP dans des paquets PPP, puis en redirigeant ces paquets PPP dans une session SSH préalablement établie *(voir figure 8.18)*.

Figure 8.18
Tunnel IP à travers SSH



L'autre extrémité réalise le cheminement inverse pour retrouver les paquets IP initiaux. L'option permettant de relayer les paquets au niveau de la pile protocolaire IP des systèmes concernés doit être activée.

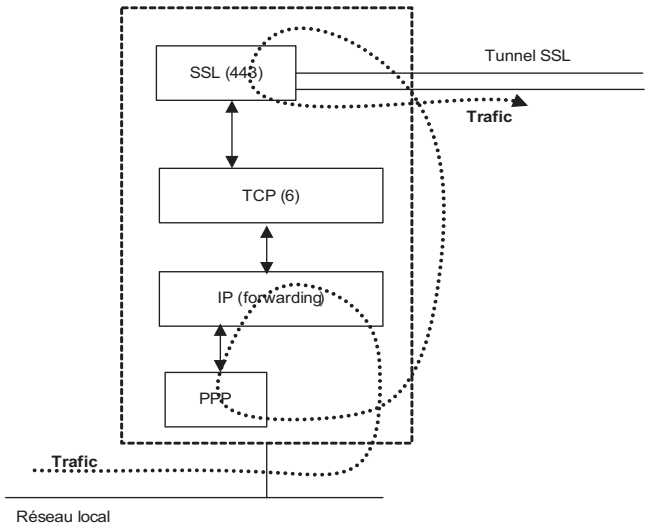
SSL (Secure Sockets Layer)

SSL permet d'établir une session client serveur sécurisée au niveau de la couche session du modèle OSI.

Le protocole PPP est généralement utilisé pour établir une interconnexion à distance à un réseau en se positionnant au niveau de la couche 2 OSI. Il permet en outre, comme précédemment, de créer un tunnel IP entre deux systèmes reliés par un réseau.

Pour créer le tunnel IP à travers SSL, il suffit d'encapsuler le trafic IP dans des paquets PPP puis de rediriger ces paquets PPP dans une session SSL *(voir figure 8.19)*.

Figure 8.19
Tunnel IP à travers SSL



Protocoles d'authentification usuels des accès distants

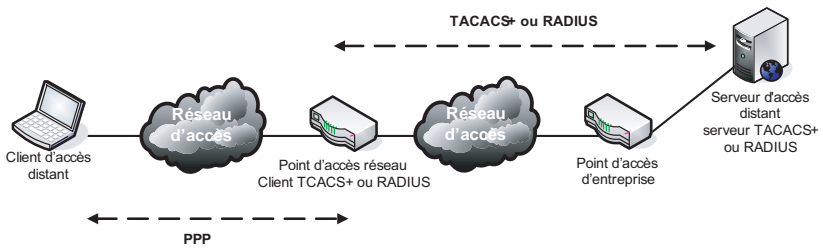
L'authentification est la première étape à réaliser lors d'un accès distant, avant les étapes d'autorisation et de journalisation des transactions.

De nombreux protocoles ont été conçus dans cette optique, tels TACACS+, RADIUS (Remote Authentication Dial-In User Server), Kerberos, etc.

Les protocoles RADIUS et TACACS+ sont les plus utilisés dans le monde des opérateurs de télécommunications pour leur simplicité d'implémentation et leur efficacité. Kerberos est surtout utilisé pour la gestion des authentifications au sein d'un système d'information.

Avant de décrire ces protocoles, il faut bien faire la différence entre l'utilisateur et le client TACACS+ ou RADIUS, car, dans la plupart des cas, le client TACACS+ ou RADIUS ne s'exécute pas sur le système de l'utilisateur. L'utilisateur se connecte généralement à distance au point d'entrée du réseau à l'aide du protocole PPP. Un client TACACS+ ou RADIUS s'exécute sur ce point d'entrée afin de relayer la demande au serveur TACACS+ ou RADIUS, comme illustré à la figure 8.20.

Figure 8.20
Authentification TACACS+ ou RADIUS



Dans ce type d'accès très répandu, les éléments de chiffrement réalisés par les protocoles TACACS+ et RADIUS sur les informations échangées ne s'appliquent que sur une partie du trafic entre le client et le serveur TACACS+/RADIUS.

TACACS+

TACACS+ est la dernière version du protocole TACACS. Développé à l'origine par BBN puis repris par Cisco, il a été étendu une première fois avec XTACACS (eXtended TACACS), compatible avec TACACS, puis par TACACS+.

TACACS+ utilise le protocole TCP et le port 49 pour son transport, contrairement à TACACS, qui s'appuie sur UDP. Il gère séparément les trois fonctions AAA (Authentication, Authorization, Accounting), c'est-à-dire l'authentification, l'autorisation et la journalisation des événements :

- **Authentification.** Pour vérifier l'identité de l'utilisateur, TACACS+ hérite des méthodes d'authentification du protocole PPP, c'est-à-dire PAP, CHAP et EAP, incluant pour la dernière méthode la possibilité d'utiliser des cartes, ou tokens. Les échanges d'authentification sont élémentaires. Ils s'appuient sur des demandes d'authentification de la part du client et des réponses d'authentification de la part du serveur. Une base de données située sur le serveur d'accès distant sur lequel s'exécute le serveur TACACS+ gère l'ensemble des utilisateurs.
- **Autorisation.** Après l'étape d'authentification de l'utilisateur, il s'agit d'assigner un profil d'utilisation ou de droits d'accès à la ressource accédée. Les échanges d'autorisation sont également élémentaires. Ils s'appuient sur des demandes d'autorisation de la part du client et des réponses d'autorisation de la part du serveur. Un profil d'autorisation sur des ressources réseau contient à la fois la liste des équipements autorisés à être accédés et celle des commandes autorisées. Il s'agit d'une option très importante pour attribuer des droits de lecture sans possibilité de modification. Les profils sont stockés sur le système hébergeant le serveur TACACS+.
- **Journalisation des événements.** Il s'agit de connaître toutes les actions menées par un utilisateur à des fins de comptabilité pour la facturation du service réseau ou à des fins d'investigation pour la gestion du réseau. Les informations disponibles sont les demandes d'authentification afin d'ouvrir une session, les fermetures de session ainsi que les actions exécutées durant une session donnée. Si plusieurs serveurs TACACS+ sont déployés, une consolidation des journaux d'activité doit être réalisée afin de corréler les événements entre eux.

Les transactions entre un client TACACS+ et un serveur TACACS+ sont authentifiées par le biais d'un secret partagé, qui n'est jamais transmis sur le réseau. Les données échangées lors de ces transactions sont chiffrées à l'aide d'une fonction XOR appliquée sur les données et une empreinte calculée à l'aide du secret partagé. Ces protections ne s'appliquent pas entre le client d'accès distant et le point d'accès réseau si c'est ce dernier qui exécute le client TACACS+.

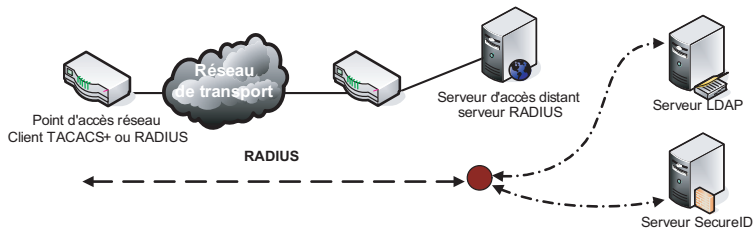
RADIUS

Créé par Livingston Enterprises, RADIUS est normalisée par les RFC 2138 et 2139 de l'IETF (Internet Engineering Task Force).

Il utilise le protocole UDP et le port 1645 — bien qu'il doive être normalement configuré sur le port 1812 — et gère les deux premières fonctions AA (Authentication, Authorization) conjointement et la troisième (Accounting) séparément.

Une possibilité native du protocole est d'agir comme relais de l'authentification vers d'autres serveurs d'authentification, qu'ils soient RADIUS ou autres (AXENT, SecureID, etc.). Cela permet d'employer un serveur d'arrière-plan pour implémenter les divers mécanismes d'authentification, tandis que le serveur authentifiant se charge de transmettre les éléments d'authentification *(voir figure 8.21)*.

Figure 8.21
Chaîne d'authentification du protocole RADIUS



Tout comme TACACS+, RADIUS hérite des méthodes d'authentification du protocole PPP. Les échanges d'authentification/autorisation s'appuient sur des demandes (de la part du client) et des réponses (de la part du serveur). Une base de données située sur le serveur d'accès distant sur lequel s'exécute le serveur RADIUS gère l'ensemble des utilisateurs RADIUS ainsi que leurs profils. L'étape préliminaire permet d'authentifier et d'autoriser un utilisateur. Il y a donc, par rapport à TACACS+, gain d'échange de messages entre le client et le serveur.

Assurer le contrôle des accès distants WI-FI

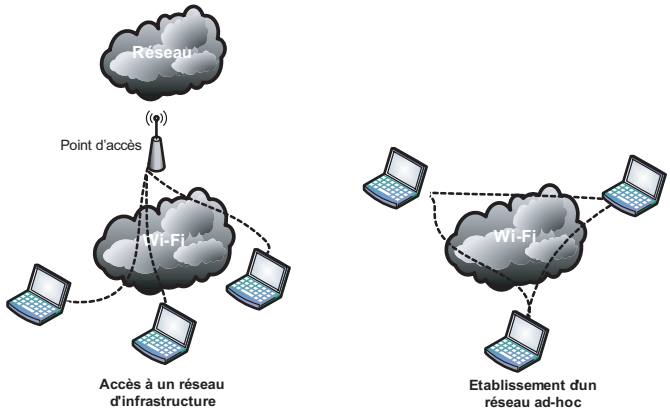
Les accès à distance à un réseau sans fil ont été définis en 1997 par le standard IEEE 802.11 . Cela couvre les couches MAC et Phy de communication entre un équipement Wireless LAN et un point d'accès ou deux équipements Wireless LAN (peer-to-peer), comme illustré à la figure 8.22.

Les éléments constituant un réseau sans fil sont les suivants :

- Point d'accès : se comporte comme une passerelle entre le réseau sans fil et le réseau filaire.
- Carte Wi-Fi : installée dans le système désirant se connecter au réseau sans fil.

Le SSID (Service Set Identifier) identifie le réseau sans fil et est configuré sur le point d'accès et le client ou appris dynamiquement par le client.

Figure 8.22
Types de topologies Wi-Fi



Côté sécurité, la norme 802.11 a défini le protocole WEP (Wired Equivalent Privacy) afin d'assurer la confidentialité et l'intégrité des données. De plus, elle a défini deux mécanismes pour gérer le contrôle d'accès et l'authentification du poste utilisateur (aucune authentification, authentification fondée sur un secret partagé).

Les principales faiblesses de sécurité du standard 802.11 sont les suivantes :

- Le vecteur d'initialisation utilisé pour le chiffrement des données est trop court et prédictible.
- La clé maître utilisée pour le chiffrement des données est trop courte.
- Il n'y a pas de gestion de clé dynamique.
- Le protocole d'authentification est trop faible (autorisation non mutuelle).

Le contrôle d'intégrité s'appuie sur un checksum linéaire.

Pour pallier les faiblesses de sécurité du protocole WEP, de nombreuses initiatives ont vu le jour au sein de la Wi-Fi Alliance afin de renforcer la sécurité de ces accès, notamment WPA (Wi-Fi Protected Access), qui implémente les fonctionnalités suivantes :

- Mécanisme de négociation d'authentification fondé sur EAP ou PSK (Pre-shared Key).
- Mécanisme de gestion et de distribution de clés TKIP (Temporal Key Integrity Protocol).
- Mécanisme d'intégrité des trames TKIP + algorithme Michael.
- Compatibilité hardware avec le parc existant : seule une migration logicielle est nécessaire.
- Compatibilité avec le WEP.
- Un nouveau protocole de chiffrement et de contrôle d'intégrité.

Le tableau 8.3 récapitule les caractéristiques des principaux standards de sécurité Wi-Fi.

Tableau 8.3 Caractéristiques des standards de sécurité Wi-Fi

	WEP	WPA	802.11i
Chiffrement	RC4	RC4	AES
Longueur de la clé	40/104 bits	128 bits	128 bits
Intégrité des données	CRC-32	Michael	CBC-MAC
Intégrité des en-têtes	Non	Michael	CBC-MAC
Contrôle des attaques par rejeu	Non	Vecteur d'initialisation	Vecteur d'initialisation
Gestion des clés	Non	802.1X	802.1X
Taille du vecteur d'initialisation	24 bits	48 bits	48 bits
Clé par paquet	Non	Oui	Possible

Les approches WPA et 802.11i renforcent l'authentification grâce aux différents types d'authentification EAP possibles *(voir tableau 8.4)*.

Tableau 8.4 Caractéristiques des authentifications EAP

Type d'EAP	Description
EAP-MD5	Adaptation du protocole CHAP de PPP. Le client s'authentifie à l'aide d'un couple login/mot de passe. Bien qu'aucun mot de passe ne transite lors de la phase d'authentification, cette méthode est vulnérable aux attaques par dictionnaire.
EAP-LEAP (LightWeight EAP)	Version améliorée d'EAP-MD5. Cette méthode est vulnérable aux attaques par dictionnaire.
EAP-TLS (Transport Level Security)	Le client et le serveur s'authentifient de manière mutuelle à l'aide de certificats X.509. Un tunnel TLS s'établit pour échanger d'autres données confidentielles.
EAP-TTLS (Tunneled TLS)	EAP-TTLS est une extension de EAP-TLS dans laquelle où une ouverture de connexion TLS est initiée entre client et serveur. Le client peut s'authentifier à l'aide d'un couple login/mot de passe protégé par un tunnel TLS préalablement établi.
EAP-PEAP (Protected EAP)	EAP-PEAP ressemble à EAP-TTLS par l'ouverture d'un tunnel TLS destiné à protéger les éléments d'authentification envoyés par le client au serveur.

Un dialogue s'établit donc entre le client, le point d'accès et le serveur d'authentification afin de valider l'accès d'un utilisateur. La figure 8.23 illustre les différentes couches réseau nécessaires au contrôle des accès des clients.

Le tableau 8.5 récapitule les caractéristiques de l'authentification côté serveur et client des différents types d'authentification EAP.
Bien que les accès Wi-Fi aient souffert de sérieuses lacunes en matière de sécurité, les derniers standards offrent maintenant de solides contre-mesures pour assurer la sécurité et le déploiement de tels accès réseau.

Il reste nécessaire de limiter l'utilisation du Wi-Fi dans un réseau d'entreprise afin de contrôler les réseaux dits *ad hoc*. Dans de tels réseaux, les éléments classiques de sécurité tels que les pare-feu, etc., pourraient être tout simplement contournés, ce qui violerait les principes de base d'une politique de sécurité réseau.

Figure 8.23
Accès à un réseau via *Wi-Fi*

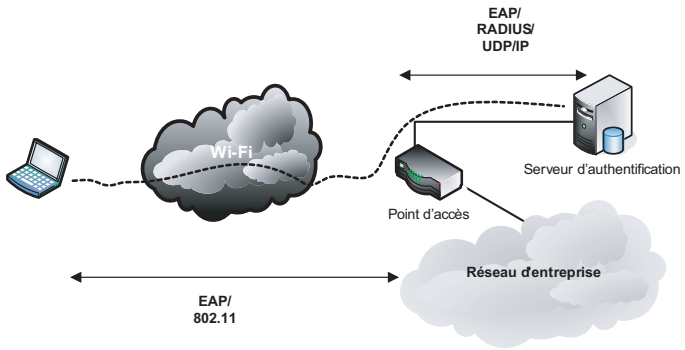


Tableau 8.5 Caractéristiques des authentifications EAP

	EAP-MD5	EAP-LEAP	EAP-TLS	EAP-TTLS	EAP-PEAP
Authentification du serveur	Aucune	Mot de passe (hash)	Certificat	Certificat	Certificat
Authentification du client	Mot de passe (hash)	Mot de passe (hash)	Certificat	Certificat, compte/ mot de passe	Certificat, compte/ mot de passe
Distribution dynamique des clés	Non	Oui	Oui	Oui	Oui

L'établissement de réseaux *ad hoc* montre aussi la nécessité de mettre en place de nouveaux outils de sécurité au sein d'une entreprise, tels que des systèmes de détection d'intrusion spécialisés dans l'écoute de réseaux Wi-Fi.

En résumé

Plusieurs méthodes d'authentification permettent de maîtriser les accès distants au réseau de l'entreprise. Ces derniers représentent une menace sérieuse pour l'entreprise si des méthodes fortes d'authentification ne sont pas mises en œuvre. Le choix et la mise en œuvre de telles méthodes nécessitent de connaître en premier lieu les besoins de sécurité de l'entreprise.

La protection des accès réseau n'est efficace que si la protection des systèmes réseau est effective et que les pirates ne peuvent pénétrer le réseau de l'entreprise par des systèmes mal protégés, évitant ainsi les mécanismes d'authentification. Le chapitre suivant détaille ces méthodes de protection des équipements réseau.