802.1X : Solution d'authentification sécurisée pour le futur réseau sans fil de l'Université Louis Pasteur

Christophe Saillard
Centre Réseau Communication, Université Louis Pasteur, Strasbourg
Christophe.Saillard@crc.u-strasbg.fr

Résumé

L'ULP souhaite déployer rapidement un réseau Wifi sur l'ensemble de ses 80 bâtiments, soit environ 900 points d'accès, à destination de différentes communautés d'utilisateurs : étudiants, enseignants et personnel. Il est donc impératif d'accompagner le déploiement physique des bornes (installation dans les bâtiments, configuration, intégration au réseau filaire) d'une infrastructure sécurisée, réalisant l'authentification des utilisateurs et assurant une confidentialité des échanges de données appropriées pour ce type de support. La solution retenue par l'ULP, pour accomplir cette tâche, est basée sur le protocole 802.1X.

Cet article présente tout d'abord une analyse détaillée de plusieurs méthodes d'authentification (basée sur des certificats, mot de passe etc.) avec EAP (Extensible Authentication Protocol).

La deuxième partie de cet article dresse un comparatif de différentes solutions 802.1X du marché, notamment celle choisie par l'ULP.

Mots clés

Wifi, Sécurité, 802.1X, EAP, RADIUS, WEP

1 Rappels sur 802.1X

Le protocole 802.1X est un standard mis au point par l'IEEE. Son but est d'autoriser l'accès physique à un réseau local après authentication depuis un réseau filaire ou sans fil.

Trois acteurs principaux interviennent dans ce mécanisme :

- Le système à authentifier (*supplicant* ou client)
- Le point d'accès au réseau local (commutateur, borne wifi etc.)
- Le serveur d'authentification

Tant qu'il n'est pas authentifié, le client ne peut pas avoir accès au réseau, seuls les échanges liés au processus d'authentification sont relayés vers le serveur d'authentification par le point d'accès. Une fois authentifié, le point d'accès laisse passer le trafic lié au client (figure 1).

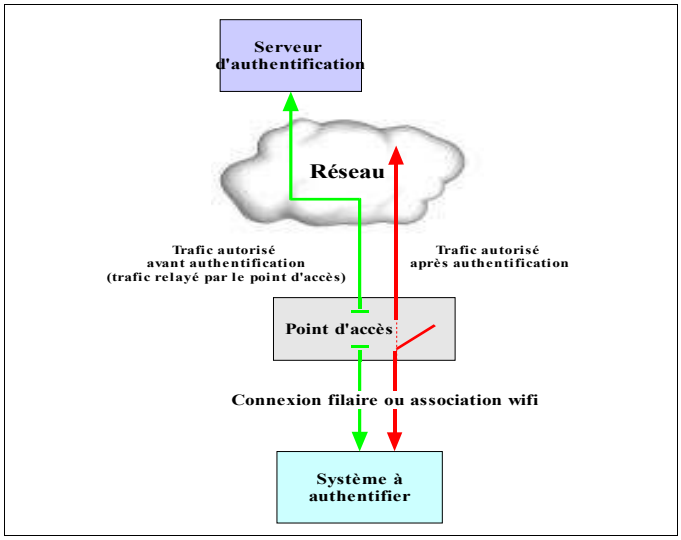

Figure 1 : Architecture d'authentification 802.1X

2 Les méthodes d'authentification de 802.1X

2.1 EAP : Extensible Authentication Protocol

Le protocole 802.1X définit l'utilisation d'EAP (Extensible Authentication Protocol, RFC 2284 [1]), mécanisme décrivant la méthode utilisée pour réaliser l'authentification. On distingue deux types de trafic EAP (figure 2):

- entre le système à authentifier et le point d'accès (support : 802.11a, b, g ou 802.3) : *EAP over LAN (EAPOL)*
- entre le point d'accès et le serveur d'authentification (de type RADIUS) : *EAP over Radius*

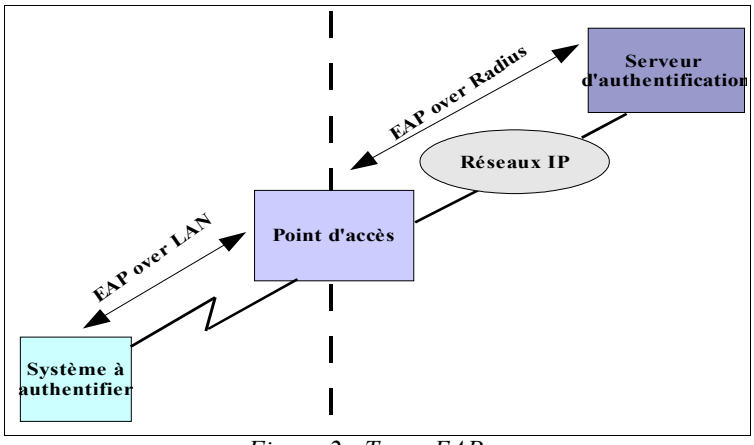

Figure 2 : Types EAP

Les messages EAP se décomposent en 4 classes :

- requêtes (du serveur vers le client)
- réponses (du client vers le serveur)
- succès
- erreur ou échec

Comme le montre la figure 3, la première étape est bien sûr l'association physique avec le point d'accès (équivalent au branchement d'un câble reliant le port d'un commutateur à l'équipement à authentifier) qui doit être réalisée préalablement à la phase d'authentification 802.1X.

La processus d'authentification est ensuite initié par l'envoi d'une requête provenant du point d'accès vers le client (EAPOL). Le client répond à la requête en y joignant un premier identifiant (nom de la machine, login, etc). Cette réponse est retransmise au serveur Radius (EAP over Radius).

À partir de ce moment, les échanges dépendent de la méthode d'authentification choisie (EAP-TLS, LEAP, etc.).

Au terme de ces échanges, le client est soit authentifié, auquel cas le point d'accès autorise le trafic du client sur le réseau, soit non-authentifié, et l'accès au réseau reste alors interdit.

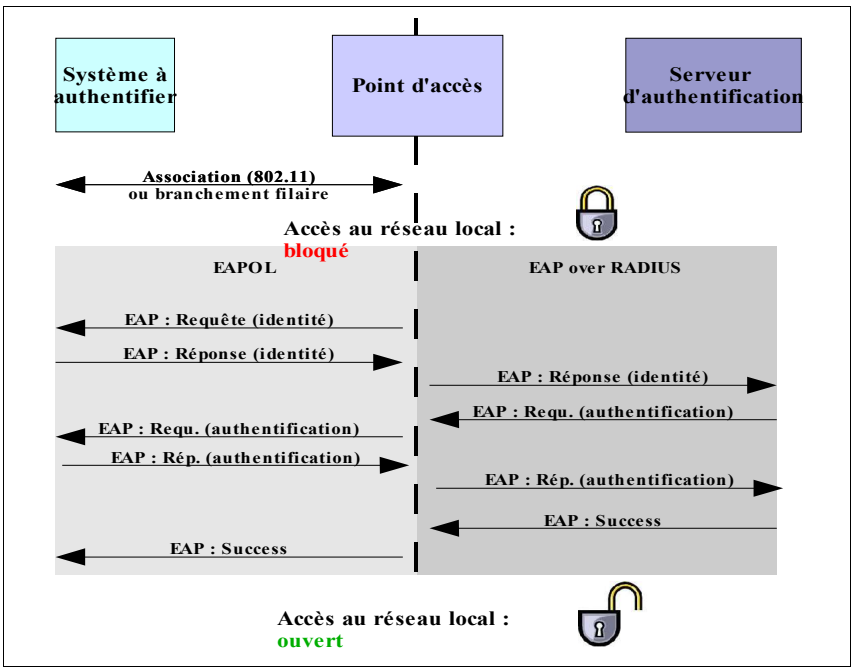

Figure 3 : Echanges EAP

2.2 Les méthodes associées à EAP

Le protocole 802.1X ne propose pas une seule méthode d'authentification mais un canevas sur lequel sont basés plusieurs types d'authentification. Ainsi, une méthode d'authentification EAP utilise différents éléments pour identifier un client :

- login / mot de passe ;
- certificat électronique ;
- biométrie ;
- puce (SIM).

Certaines méthodes combinent plusieurs critères (certificats et login/mot de passe etc.)

En plus de l'authentification, EAP gère la distribution dynamique des clés de chiffrement (WEP).

Le reste de cet article décrit plus en profondeur les méthodes suivantes :

- **EAP-TLS** [2] : authentification mutuelle entre le client et le serveur Radius par le biais de certificats (côté client et côté serveur) ;
- **EAP-TTLS** [3] et **EAP-PEAP** [4] : authentification mutuelle du client et du serveur Radius par le biais d'un certificat côté serveur, le client peut utiliser un couple login/mot de passe ;
- **EAP-MD5** : pas d'authentification mutuelle entre client et le serveur Radius, le client s'authentifie par mot de passe ;
- **EAP-LEAP** : cas particulier, méthode propriétaire de Cisco.

2.2.1 EAP-TLS (Transport Layer Security)

Comme d'autres protocoles (SMTP-TLS, IMAP-TLS, HTTPS, etc.), EAP s'appuie sur TLS pour proposer une authentification sécurisée. Cette méthode s'appuie sur les certificats électroniques. Ainsi, chaque partie (serveur et client) doit posséder un certificat pour prouver son identité.

L'utilisation de certificats possède des avantages et des inconvénients. Ils sont souvent considérés comme plus sûrs que les mots de passe, cependant les opérations de gestion qu'ils engendrent peuvent se réveler fastidieuses (création, suppression, listes de révocation etc.) et l'existence d'une infrastructure de gestion de clés (IGC) est requise. La distribution des certificats aux clients est une contrainte qu'il ne faut pas négliger.

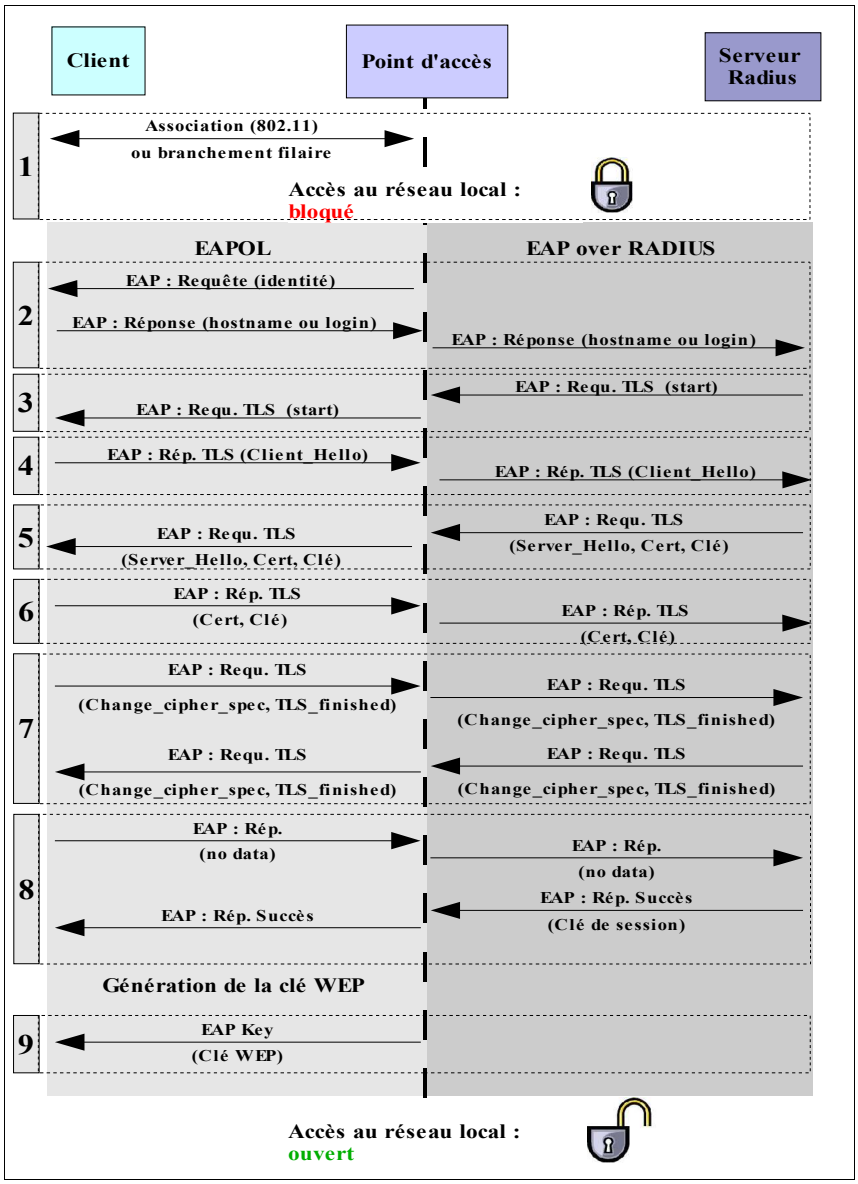

Figure 4 : Diagramme d'échanges EAP-TLS

Les explications suivantes se réfèrent aux étapes numérotées dans la figure 4.

1) Le client s'associe physiquement au point d'accès.

2) Le point d'accès envoie une requête d'authentification au client. Le client répond avec son identifiant (nom de machine ou login), ce message est relayé par le point d'accès vers le serveur Radius.

3) Le serveur Radius initie le processus d'authentification TLS par le message *TLS start*.

4) Le client répond avec un message *client_hello*, qui contient :
- des spécifications de chiffrement vides en attendant qu'elles soient négociées entre le client et le serveur ;
- la version TLS du client ;
- un nombre aléatoire (défi ou *challenge*) ;
- un identifiant de session ;
- les types d'algorithmes de chiffrement supportés par le client.

5) Le serveur renvoie une requête contenant un message *server_hello* suivi
- de son certificat (*x509*) et de sa clé publique ;
- de la demande du certificat du client ;

- d'un nombre aléatoire (défi ou *challenge*) ;
- d'un identifiant de session (en fonction de celui proposé par le client).

Le serveur choisit un algorithme de chiffrement parmi ceux qui lui ont été proposés par le client.

6) Le client vérifie le certificat du serveur et répond avec son propre certificat et sa clé publique.

7) Le serveur et le client, chacun de son côté, définissent une clé de chiffrement principale utilisée pour la session. Cette clé est dérivée des valeurs aléatoires que se sont échangées le client et le serveur. Les messages *change_cipher_spec* indiquent la prise en compte du changement de clé. Le message *TLS_finished* termine la phase d'authentification TLS (*TLS handshake*), dans le cas d'EAP-TLS la clé de session ne sert pas à chiffrer les échanges suivants.

8) Si le client a pu vérifier l'identité du serveur (avec le certificat et la clé publique), il renvoie une réponse EAP sans donnée. Le serveur retourne une réponse *EAP success*.

9) La clé de session générée en (8) est réutilisée par le point d'accès pour créer une clé WEP qui est transmise au client, dans le cas où il s'agit d'une station Wifi. La clé de session est valide jusqu'à ce que le client se déconnecte ou que son authentification expire, auquel cas il doit s'identifier à nouveau.

Le tunnel TLS créé lors de la création de la clé de session n'est donc pas exploité. Seul le *TLS Handshake* est utilisé, il permet l'authentification mutuelle des deux parties.

EAP-TLS est une méthode d'authentification performante. Seul les problèmes liés aux IGC peuvent dissuader de l'utilisation de cette méthode.

2.2.2 EAP-TTLS (Tunneled TLS) et EAP-PEAP (Protected EAP)

Ces deux méthodes sont assez similaires. Elles s'appuient sur la confidentialité proposée par l'encapsulation dans un tunnel pour réaliser une authentification via login/mot de passe ou *token-card*.

On distingue deux phases d'authentification :

- Première phase : identification du serveur par le client en utilisant un certificat (validé par une autorité de certification)
- Deuxième phase : identification du client par le serveur par login/password

À l'issue de la première phase, le tunnel TLS chiffré s'établit, garantissant une grande confidentialité des échanges pour la phase 2 où le client transmet ses éléments d'authentification (login/password) via *CHAP*, *PAP*, *MS-CHAP* ou *MS-CHAPv2* pour EAP-TTLS et *MS-CHAPv2*, *token-card* ou certificat (similaire à EAP-TLS) pour EAP-PEAP.

La différence principale entre EAP-PEAP et EAP-TTLS vient de la manière d'encapsuler les échanges lors de la deuxième phase. Pour EAP-PEAP, les données échangées entre le client et le serveur au travers du tunnel TLS sont encapsulées dans des paquets EAP. EAP-TTLS utilisent des AVP (*Attribute-Values Pairs*) encapsulées dans des paquets EAP-TTLS, le format AVP d'EAP-TTLS est compatible avec le format AVP de Radius, ce qui simplifie les échanges entre le serveur EAP-TTLS et le serveur Radius qui contient les informations relatives aux utilisateurs, dans le cas où les informations ne sont pas directement stockées sur le serveur EAP-TTLS. La méthode EAP-PEAP ne peut-être utilisée qu'avec un serveur Radius supportant EAP (figure 5). EAP-TTLS est plus souple, il est toujours nécessaire de dialoguer avec un serveur EAP, cependant ce serveur peut retransmettre directement la requête auprès d'un serveur Radius ne gérant pas EAP (figure 6).

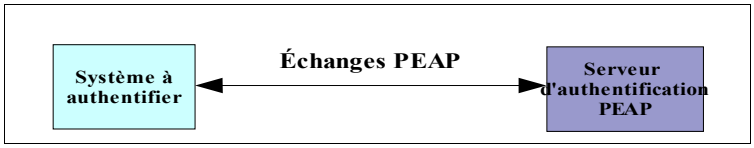

Figure 5 : Echanges EAP-PEAP

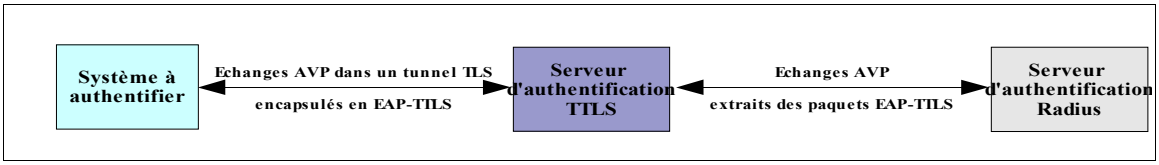

Figure 6 : Echanges EAP-TTLS

L'avantage présenté par ces deux méthodes vient du fait que le client peut être authentifié par mot de passe, on supprime donc la complexité de gestion liée aux certificats caractéristique de EAP-TLS, tout en proposant une authentification mutuelle.

PEAP est proposé nativement dans Windows XP et Windows 2000, ce qui peut grandement faciliter son déploiement. Par ailleurs, l'implémentation de EAP-PEAP dans les clients d'authentification proposés par Cisco n'est pas compatible avec celle de Microsoft. Ainsi, on ne pourra pas s'authentifier avec EAP-PEAP depuis un client natif Windows sur un serveur Radius Cisco de type ACS.

EAP-TTLS et PEAP s'adressent principalement aux sites ne disposant pas d'IGC. De plus, il est tout à fait possible d'utiliser les informations stockées dans un annuaire LDAP relié au serveur RADIUS pour proposer un identifiant unique pour toutes les applications.

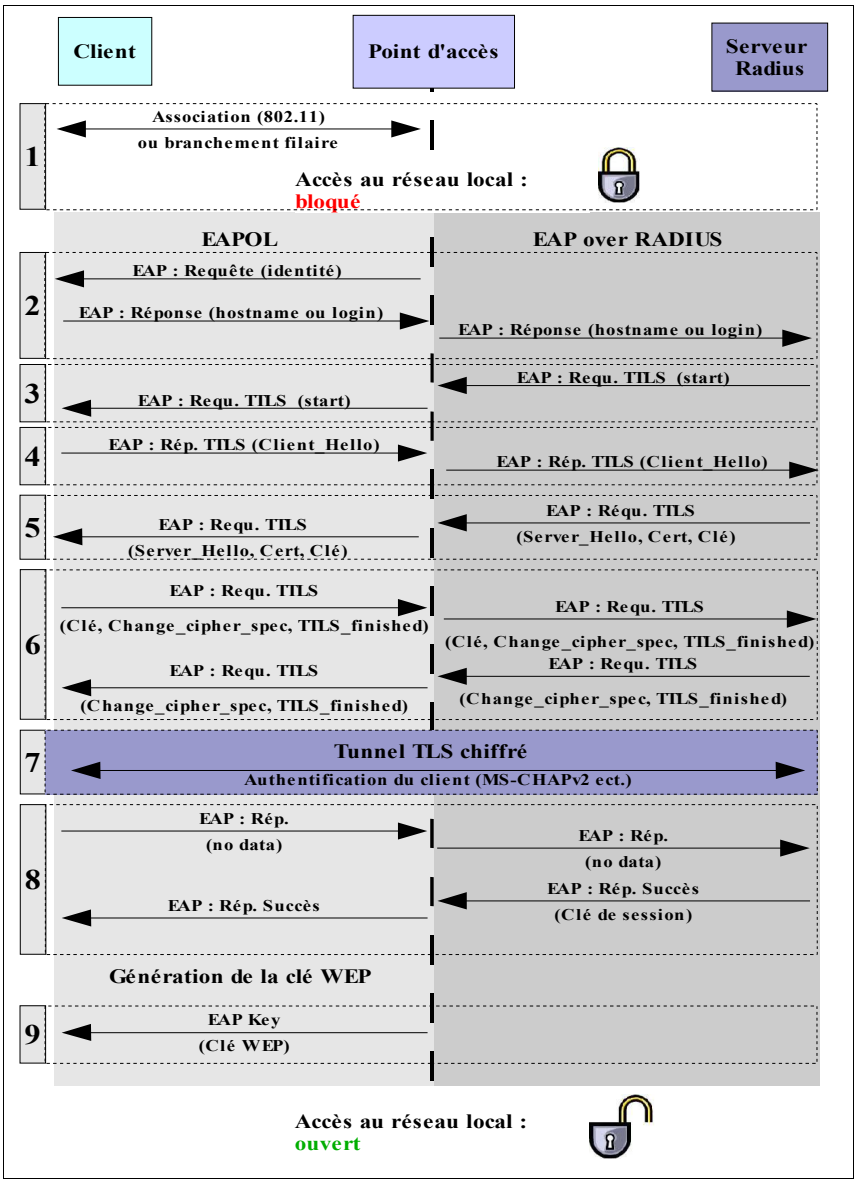

Figure 7 : Diagramme d'échanges EAP-TTLS ou EAP-PEAP

Les explications suivantes se réfèrent aux étapes numérotées dans la figure 7.

1 à 5) Les échanges sont presque similaires à EAP-TLS. Le client authentifie le serveur par l'intermédiaire d'un certificat (étape 5).

6) Cette étape diffère légerement d'EAP-TLS car le client n'a pas besoin de fournir de certificat, la clé qui sert à chiffrer la session peut donc être créée directement. À la fin de cette étape, le *TLS handshake* est terminé, les échanges suivants seront donc chiffrés par la clé de session.

7) En effet, l'établissement d'un tunnel TLS permet de chiffrer les échanges, le client fournit donc ses identifiants (login/mot de passe) au serveur en utilisant par exemple *MS-CHAPv2*.

8 et 9) Similaires à EAP-TLS

EAP-TTLS et EAP-PEAP sont des méthodes très proches et l'utilisation d'un tunnel TLS chiffré leur confère un bon niveau de confidentialité. EAP-PEAP présente l'avantage d'être supporté nativement par Windows XP et 2000. EAP-TTLS permet une meilleure interopérabilité avec les serveurs Radius ne supportant pas EAP.

2.2.3 EAP-MD5

EAP-MD5 ne propose pas d'authentification mutuelle, le client s'authentifie simplement en fournissant un couple login/mot de passe.

Le problème majeur de cette méthode réside dans le fait que les échanges ne sont pas chiffrés. En outre, EAP-MD5 ne gère pas la distribution dynamique des clés WEP.

Le seul avantage de cette méthode est la simplicité : il est relativement facile de mettre en place une structure d'authentification basée sur cette méthode. Celle-ci est d'ailleurs beaucoup utilisé pour des réseaux filaires où la contrainte lié au chiffrage des échanges est moins forte que pour les réseaux Wifi.

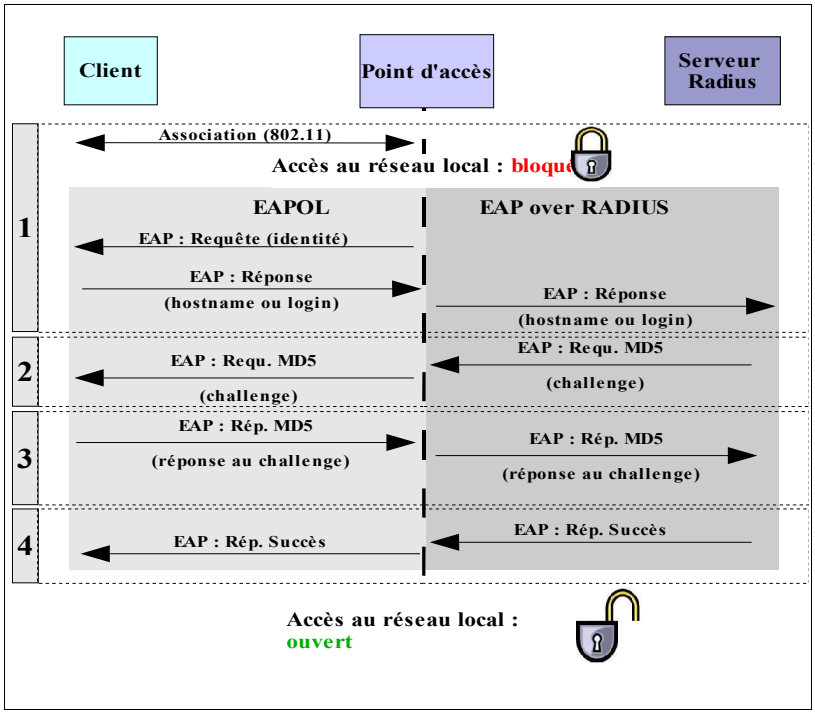

Figure 8 : Diagramme d'échanges EAP-MD5

Les explications suivantes se réfèrent aux étapes numérotées dans la figure 7.

1) Après l'association et la phase EAP standard de demande d'identification, le serveur émet une requête EAP-MD5 sous forme d'un texte de défi ou *challenge text* (2).

3) Le client doit répondre à cette requête en chiffrant le défi avec son mot de passe.

4) Le serveur chiffre le défi de son côté en utilisant le mot de passe du client stocké dans sa base. Si le résultat coïncide, le client est authentifié.

Il est très important de noter que les échanges sont non chiffrés. Le *challenge text* et son résultat chiffré transitent en clair sur le réseau.

Cette méthode est vulnérable aux attaques de types Dictionnaire (pas de notion de session, attaque brute possible), *Man In the Middle*, *session hijacking* .

Des améliorations permettent de chiffrer les échanges entre le client et le serveur (utilisation de tunnel chiffré), mais le fait qu'EAP-MD5 n'offre pas la possibilité de générer dynamiquement des clés WEP le rend inutilisable pour les réseaux sans-fil.

2.2.4 Cisco LEAP (Lightweight EAP)

Cisco a développé sa propre méthode EAP de manière à pouvoir proposer une solution complète (cartes clients, points d'accès, serveur Radius). Malgré tout, les équipements Cisco supportent d'autres types d'authentification (PEAP, EAP-TLS etc.).

Du point de vue de l'administrateur, le fait d'avoir une solution complète est un avantage indéniable. Cependant, il est difficile de contraindre tous les utilisateurs à s'équiper avec des cartes et des points d'accès d'un seul constructeur. Heureusement, il existe des logiciels clients permettant de s'authentifier en utilisant LEAP avec des cartes d'autres constructeurs.

Par ailleurs, LEAP présente quelques défaillances. Tout d'abord, la clé utilisée entre le client et le point d'accès est dérivée du login et du mot de passe stockés sur le serveur Radius. La méthode utilisée dans ce cas est *MS-CHAPv1*, connue pour être vulnérable. Ensuite, Les échanges EAP ne sont pas chiffrés, le login passe en clair, seul le mot de passe est protégé par le hachage *MS-CHAPv1*.

2.3 Tableau récapitulatif

Type EAP	*Gestion dynamique des clés WEP*	*Re-Authentification automatique*	*Méthode d'authentification*	*Remarques*
EAP-MD5	NON	NON	Login/Password	• Facile à implémenter • Supporté par beaucoup de serveur • Utilise les mots de passe en clair • Pas d'authentification mutuelle
EAP-TLS	OUI	OUI	Certificat	• Utilisation de certificats pour le serveur et les clients • Solide mais plus compliqué à gérer à cause des certificats • Authentification mutuelle entre le serveur et le client
EAP-LEAP	OUI	OUI	Login/Password	• Solution propriétaire
EAP-TTLS	OUI	OUI	- Login/Password - Certificat	• Création d'un tunnel TLS sûr • Supporte PAP, CHAP, MS-CHAP, MS-CHAPv2 • Certificat obligatoire côté serveur, optionnel côté client • Authentification mutuelle
EAP-PEAP	OUI	OUI	- Login/Password - Certificat	• Similaire à EAP-TTLS • Création d'un tunnel TLS sûr • Authentification mutuelle

Nous avons pu voir que le choix de la méthode d'authentification a un fort impact sur la gestion du système. Par exemple, l'utilisation d'EAP-TLS implique l'existence d'une infrastructure de gestion de clés. Le déploiement d'une telle infrastructure est en soi un projet d'ampleur qu'il ne faut pas négliger. Si l'IGC existe, il faut l'utiliser ; dans le cas contraire, on préférera une solution basée sur EAP-PEAP ou EAP-TTLS, qui nécessite un seul certificat pour le serveur. Le client s'authentifiera par login /mot de passe stockés sur le serveur d'authentification (RADIUS). Il est possible d'utiliser les informations issues d'un annuaire de type LDAP en le connectant au serveur RADIUS.

Pour utiliser les méthodes EAP décrites dans cet article, il est absolument indispensable de mettre en place une architecture d'authentification. La partie suivante présente plusieurs solutions d'architectures que l'on peut trouver actuellement sur le marché.

3 Les solutions 802.1X du marché

3.1 Les contraintes de l'ULP

La solution utilisée par l'ULP pour réaliser l'authentification des utilisateurs du réseau sans fil de l'ULP devait respecter plusieurs contraintes.

La plus importante est sans conteste le fait que le client d'authentification intégré à la solution devrait pouvoir fonctionner dans un milieu hétérogène composé de :

- plates-formes PC, Mac, Palm, PocketPc etc.
- systèmes d'exploitation de type Windows XP/2000/98, Linux, MacOS X etc.
- de cartes clientes de marques variées

En effet, la diversité des publics visés (étudiants, enseignants, personnels) implique que la solution doit prévoir un maximum de cas.

Par ailleurs, il était souhaitable que la solution propose le support d'un grand nombre de méthodes EAP. Enfin, l'homogénéïté et la facilité de déploiement de la solution étaient des points à vérifier.

3.2 Les solutions testées

Trois solutions ont été testées.

3.2.1 Solution « libre »

- **Serveur Radius** : *FreeRADIUS sous Linux* [5]
- **Client d'authentification** : *Xsupplicant sous Linux* [6]*, Natif Windows XP/SP1*

La première solution, basée sur les logiciels libres, a consisté en l'installation de FreeRADIUS avec un module d'authentification EAP-TLS. Côté client, nous avons opté pour Xsupplicant sous Linux, et la prise en charge native d'EAP-TLS sous Windows XP nous a permis de nous affranchir de l'installation d'un client pour ce système.

Cette solution donne satisfaction, mais l'état d'avancement du serveur **FreeRADIUS** et du client **Xsupplicant** ne permet pas, à l'heure actuelle, d'envisager un déploiement sérieux basé sur cette architecture.

D'autre part, dans un souci de cohérence, et pour faciliter l'exploitation, il est préférable d'avoir un seul type de client d'authentification sur l'ensemble des plate-formes. Cette solution ne répond donc pas à cette attente.

Par ailleurs, la version de FreeRADIUS testée ne permettait qu'une authentification EAP-TLS, nécessitant l'existance d'une infrastructure de gestion de clé. Pour les besoins du test, une « mini » IGC a été installée, mais l'ULP ne possède pas encore une telle infrastructure. Une prise en charge de PEAP aurait donc été apréciable.

3.2.2 Solution « Cisco »

- **Serveur Radius** : *ACS 3.1 sous Windows 2000*
- **Client d'authentification :** *ACU sous Windows et sous Linux*

ACS et ACU prennent en charge la plupart des méthodes EAP actuelles. Ces 2 logiciels sont assez simples à configurer. Le serveur d'authentification ACS ne fonctionne malheureusement que sous Windows. De plus, son prix est relativement élevé.

ACU ne fonctionne qu'avec les cartes Cisco et sur un nombre limité de plate-formes (Windows 2000/XPet Linux avec un kernel 2.4).

3.2.3 Solution « MeetingHouse » [7]

- **Serveur Radius** : *Aegis Premium Server sous Linux*
- **Client d'authentification :** *Aegis Client sous Windows et sous Linux*

Cette solution est la plus pertinente pour les besoins de l'ULP :

- Elle fonctionne sur la majorité des plate-formes actuelles avec un client universel (Windows NT 4.0 avec Service Pack 6a, 98, 98SE, ME,CE.net, 2000, XP, MacOS 10.2.x, Linux avec kernel 2.4, Solaris 8, PocketPc 2002, Palm Tungsten), le serveur tourne sous Linux, Windows 2000 et Solaris 8.
- Elle prend en charge la plupart des méthodes EAP existantes
- Elle permet de s'authentifier avec la méthode LEAP avec d'autres cartes que Cisco

L'ULP a donc retenu cette offre. Elle s'est dotée d'une version du serveur *Aegis Premium* pour Linux et de 5000 licences pour le client *Aegis*.

Conclusion

Le choix d'une méthode et d'une structure d'authentification n'est pas aisé devant la quantité de solutions offertes. Ce choix peut être grandement influencé par les informations relatives aux utilisateurs pré existantes.(IGC, annuaire LDAP, etc.).

En ayant connaissance des faiblesses de sécurité des réseaux de type Wifi et au vu de l'essor important de ce type de matériel, il est probable que le marché des serveurs d'authentification va prendre de l'importance. Ainsi, depuis les tests, certains produits ont déjà beaucoup évolué pour prendre en charge davantage de méthodes d'authentification et de plateformes (FreeRADIUS supporte désormais EAP-TTLS et LEAP). Cependant, sur le segment de la sécurité des réseaux Wifi, d'autres solutions restent envisageables notamment celles basées sur les VPN (*Virtual Private Network*).

Le niveau de sécurité proposé par 802.1X est correct mais il ne permet pas de résoudre les problèmes liés aux faiblesses de WEP. Ainsi, pour proposer une architecture vraiment sûre il faudra utiliser d'autres techniques de chiffrement comme WPA (*Wifi Protected Access*) et attendre les avancées proposées par 802.11i (prévu pour fin 2003).

La relative jeunesse de tous ces protocoles, et des réseaux Wifi en général, ne permettent pas encore de garantir une pérénité de la solution retenue. Malgré tout, il est préférable de prendre le risque d'opter pour une solution plutôt que d'attendre et de laisser son réseau sans fil sans protection.

Références

[1] RFC2284 EAP, *http://www.ietf.org/rfc/rfc2284.txt*

[2] RFC2716 EAP-TLS, *http://www.ietf.org/rfc/rfc2716.txt*

[3] Internet Draft sur EAP-PEAP, *http://www.globecom.net/ietf/draft/draft-josefsson-pppext-eap-tls-eap-02.html*

[4] Internet Draft sur EAP-TTLS, *http://www.ietf.org/internet-drafts/draft-ietf-pppext-eap-ttls-03.txt*

[5] Serveur RADIUS, *http://www.freeradius.org*

[6] Client 802.1X libre, *http://www.open1x.org*

[7] MeetingHouse, *http://www.mtghouse.com*