Afrique SCIENCE *02(1) (2006) 22 - 38*
ISSN 1813-548X

Architecture d'une solution de sécurité dans un réseau sans fil basé sur la norme IEEE 802.1x

Souleymane OUMTANAGA[1,*] **et Jimmy KOURAOGO**[2]

[1]*LABoratoire des Technologies de l'Information et de la Communication (LABTIC), INPHB, BP 1093 Yamoussoukro, Côte d'Ivoire*
[2]*01 BP 317 Abidjan 01, Côte d'Ivoire, courriel :* jkouraogo@yahoo.fr

(Reçu le 19 Septembre 2005, accepté le 27 Décembre 2005)

* Correspondance, courriel : oumtana@nic.ci

Résumé

La norme 802.11b est devenue une référence en matière de communication informatique sans fil, mais son système de sécurité présente plusieurs vulnérabilités. C'est pourquoi de nombreux cryptages et autres solutions techniques sont venus en aide au 802.11b. Même si ces solutions ne résolvent pas de façon définitive le problème de sécurité, il est à noter q'elles l'améliorent fortement. Nous proposons d'aborder le problème de sécurité rencontré dans les réseaux 802.11b par l'étude des différentes évolutions concernant les systèmes de sécurité et présentons l'architecture d'une solution de sécurité basée sur l'utilisation d'un portail web et de la norme IEEE 802.1x.
A travers cet article nous montrons qu'il est possible de mettre en oeuvre une architecture qui puisse améliorer la sécurité des réseaux sans fil 802.11b.

Mots-clés : *IEEE 802.11b, services IP, mobilité, serveur Radius, portail web, authentification, EAP-TLS, TKIP, WPA, WPA2.*

Abstract

Architecture of a safety solution in a Wireless network based on the standard IEEE 802.1x

The standard 802.11b became a reference as regards data-processing communication wireless, but its system of safety presents several vulnerabilities. This is why of many encodings and other technical solutions came to assistance of the 802.11b. Even if these solutions do not resolve in a final way the problem of safety, it should be noted that they strongly improve it. We propose to tackle the problem of safety met in the networks 802.11b by the study of the various evolutions concerning the systems of safety and we

present the architecture of a solution of safety based on the use of a gate Web and standard IEEE 802.1x.
Through this architecture we show that it is possible to implement an architecture which can improve safety in wireless LAN 802.11b.

Keywords: *IEEE 802.11b, Services IP, roaming, Radius server, gate web, authentication, EAP-TLS, TKIP, WPA, WPA2.*

1. Introduction

La norme 802.11b [1] prévoit le protocole WEP (Wired Equivalent Privacy) [2] pour sécuriser les échanges par l'utilisation d'un secret partagé par tous les équipements ayant le droit de se connecter. Si ce secret partagé est divulgué ou facilement repérable, c'est la sécurité du réseau dans son ensemble qui est mise en jeu. En effet le WEP utilise l'algorithme RC4 pour chiffrer les données sur l'AP (Acces Point) et sur les clients avec une clé de 64 ou 128 bits. Fluhrer, Mantin et Shamir [3] ont montré qu'il y avait des failles dans cet algorithme dont une vulnérabilité ayant trait à la génération de la chaîne pseudo-aléatoire à partir du secret partagé (Key Scheduling Algorithm).

Grâce à ses failles, il est donc possible (Wardriving) [4] de détourner une connexion réseau à son avantage et éventuellement pouvoir utilisé Internet gratuitement par usurpation de l'identité du point d'accès et envoi d'un paquet trafiqué au client sans fil lui faisant croire que la connexion est terminée. Une attaque de type MITM (Man In The Middle) est possible en se faisant passer pour le point d'accès vis-à-vis du client sans fil, mais il n'est en fait qu'un relais entre le client et le point d'accès, ce qui lui permet de capter toute la " conversation ".

Le but de ce document est de présenter une architecture d'une solution de sécurité dans un réseau sans fil, basée sur la norme IEEE 802.1X [5,16], les fonctions EAP-TLS (Extensible Authentication Protocol - Transport Layer Security) [6] et le standard WPA (Wi-Fi Protected Access) [7]. Cette architecture présente les différents équipements et logiciels nécessaires, leur emplacement de même que les types d'utilisateurs à gérer. Il est à noter que tous les tests ont été réalisés au sein du LABTIC (LABoratoire des Technologies de l'Information et de la Communication) et qu'une phase de déploiement au sein d'un établissement est en projet.

Cette architecture doit donc offrir les fonctionnalités suivantes :
- Une authentification fiable et réciproque du client, du point d'accès sans fil et du serveur d'authentification;

- Un processus d'autorisation qui détermine qui est autorisé et qui n'est pas autorisé à accéder au réseau sans fil;
- Un cryptage de haut niveau du trafic du réseau sans fil;
- Une gestion sécurisée des clés de cryptage;
- Une résilience aux attaques de refus de service.

Afin de répondre aux exigences relevées ci-dessus, nous nous appuyons sur les outils suivants :
- Utilisation d'un portail web en l'occurrence NoCat [8] dont l'objectif est de rediriger toutes les connections http des clients vers une passerelle (gateway) en utilisant le protocole sécurisé https ;
- Authentification de ces clients grâce à l'utilisation du standard 802.1x normalisé par l'IEEE, basé sur une authentification par ports et dont l'objectif est d'autoriser l'accès physique à un réseau local (802.5, 802.11, etc.) après une phase d'authentification ;
- Enfin l'utilisation du WPA afin d'assurer une gestion sécurisée des clés de cryptage.

Après cette première partie introductive, la partie 2 de cet article présente les différentes évolutions en matière de sécurité au sein des réseaux 802.11b. La partie 3 est consacrée à la présentation de la solution de sécurité que nous proposons. La plate-forme d'expérimentation et son implémentation sont abordées dans la partie 4. Enfin, la partie 5 sera consacrée à la conclusion dans laquelle nous présentons les perspectives de la nouvelle norme 802.11i [9], qui devrait améliorer encore plus la sécurité et les contraintes liées à la 802.11b.

2. Les différentes évolutions de la sécurité des réseaux sans fil

Jusqu'a ce jour, il existe 4 générations de sécurité Wi-Fi :
- Le WEP [2]
- Le 80.1x ou WEP2 [5]
- Le WPA [7]
- Le 802.11i [9]

2-1. Le cryptage WEP

En plus de la sécurité basée sur le filtrage d'adresse MAC, tout équipement Wi-Fi dispose par défaut, du protocole de cryptage WEP. Pour le mettre en place, il suffit juste de l'activer et de le configurer.

Le principe du WEP consiste à définir dans un premier temps une clé secrète de 40 ou 128 bits. Cette clé secrète doit être déclarée au niveau du point d'accès et des clients. La clé sert à créer un nombre pseudo-aléatoire d'une longueur égale à la longueur de la trame.

Chaque transmission de données est ainsi chiffrée en utilisant le nombre pseudo-aléatoire comme masque grâce à un «OU Exclusif» entre le nombre pseudo-aléatoire et la trame.

La clé de session partagée par toutes les stations est statique, c'est-à-dire que pour déployer un grand nombre de stations Wi-Fi, il est nécessaire de les configurer en utilisant la même clé de session. Ainsi, la connaissance de la clé est suffisante pour déchiffrer les communications. De plus, 24 bits de la clé servent uniquement pour l'initialisation, ce qui signifie que seuls 40 bits de la clé de 64 bits servent réellement à chiffrer et 104 bits pour la clé de 128 bits.

Ainsi, d'une part, il est aisé pour n'importe quel utilisateur disposant d'outil d'espionnage [10], à partir de simples écoutes du réseau, de déterminer tous vos paramètres de sécurité, en quelques heures seulement. D'autre part, ce protocole ne gère pas les attaques de type « déni de service » (DoS). Aussi, n'importe quel individu mal intentionné peut-il saturer le réseau et le rendre non opérationnel pendant un bon moment. Le WEP n'est pas du tout sécurisé et donc sensible à des attaques. Pour pallier ces problèmes, plusieurs solutions ont été envisagées.

2-2. Le 802.1x ou WEP2

Ce standard repose sur le protocole d'authentification EAP (Extensible Authentication Protocol) [11], défini par l'IETF (Internet Engineering Task Force) [12], dont le rôle est de transporter les informations d'identification des utilisateurs. Il offre une légère amélioration de certains problèmes de sécurité rencontrés avec le WEP notamment au niveau des clés où on assistait à une gestion et une création dynamique des clés à utiliser et au niveau de l'authentification, qui se fait désormais à l'aide du protocole RADIUS (Remote Authentication Dial - In User Service) [13].

Cependant, il a été démontré, par deux chercheurs de l'université de Maryland [14], que l'authentification de l'utilisateur à l'aide du 802.1x basique présentait deux gros problèmes et n'est donc pas totalement sûre. Celle-ci ne garantit pas la protection contre l'écoute du trafic d'authentification, l'attaque du type MITM (Man In The Middle) et le DOS (Deny Of Service).

2-3. Le WPA

Le standard WPA (Wi-Fi Protected Access) proposé par la Wi-Fi Alliance [7] regroupe toutes les normes précédentes ainsi que, entre autres, un protocole standardisé de gestion des clés, appelé TKIP (Temporal Key Integrity Protocol). Le protocole TKIP permet la génération aléatoire de clés et offre la possibilité de modifier la clé de chiffrement

plusieurs fois par seconde, pour plus de sécurité. Il s'agit d'une amélioration considérable en matière de sécurité des solutions de réseau local sans fil, reconnue par la majorité des spécialistes.

Il peut être implémenté suivant deux méthodes :

- Le mode PSK (Preshared Key): version restreinte du WPA, mis en oeuvre dans les petits réseaux où une même clé de chiffrement est déployée dans l'ensemble des équipements, ce qui évite la mise en place d'un serveur RADIUS.
- Le mode entreprise en association avec un serveur d'authentification ; ce qui assure une plus grande sécurité au réseau.

Le WPA (dans sa première monture) ne supporte que les réseaux en mode infrastructure, ce qui signifie qu'il ne permet pas de sécuriser des réseaux sans fil d'égal à égal (mode ad hoc).

Avec le WPA on assiste au passage de l'IV (Vecteur d'Initialisation) de 24 à 48 bits grâce à l'usage de TKIP. Un contrôle d'intégrité de messages MIC (*Message Integrity Check),* baptisé aussi Michael, permet de savoir si les paquets ont été ou non altérés. L'inconvénient du WPA est qu'il reste lié à l'utilisation de l'algorithme de chiffrement RC4 sur lequel TKIP est basé.

2-4. Le 802.11i ou WPA2

Le 802.11i [9] ratifié le 24 juin 2004, fournit une solution de sécurisation poussée des réseaux Wi-Fi. Il s'appuie sur l'algorithme de chiffrement TKIP, comme le WPA, mais supporte également l'AES (Advanced Encryption Standard) [15], beaucoup plus sûr. La Wi-Fi Alliance a ainsi créé une nouvelle certification, baptisée WPA2, pour les matériels supportant le standard 802.11i.

Contrairement au WPA, le WPA2 permet de sécuriser aussi bien les réseaux sans fil en mode infrastructure que les réseaux en mode ad hoc.

Le WAP2 peut être utilisé avec :

- Le TKIP (Tempory Key Integrety Protocol) dont le principe est d'utiliser des clés dynamiques et uniques pour chaque équipement.
- Le CCMP : composé de CTR (Counter Mode encryptage), CBC (Cipher Block Chaining) et MAC (Message Authentication Code).

L'inconvénient de WPA2 est qu'il nécessite obligatoirement l'utilisation de nouveaux équipements dotés d'une puissance de calcul capable d'exécuter les opérations AES. C'est peut-être ce qui a fait dire à Brian Gimm, le Directeur marketing de la Wi-Fi Alliance [7], lors de l'édition du *CeBIT 2003*, à Hanovre (Allemagne), que *« WPA est appelé à durer »* .

A partir de cette brève présentation de l'évolution de la sécurité dans les réseaux Wi-Fi, l'on peut donc, dans l'optique d'un renforcement de la sécurité et dans l'attente d'une vulgarisation de nouveaux équipements supportant la norme 802.11i, retenir ce qui suit :
- Utilisation de la norme 802.1x qui, associé aux fonctions EAP-TLS [6], permet, sur le contrôle des accès au réseau de répondre à certaines des exigences définies dans l'introduction (*Paragraphe 2, ligne 15)* telles que l'authentification mutuelle du client, du point d'accès et du serveur Radius ;
- Utilisation du standard WPA pour une meilleure gestion des clés de cryptage, un contrôle d'intégrité de messages (MIC).

Tableau 1 : *Tableau récapitulatif*

	WEP	**WEP+802.1x**	**WPA (mode PSK)**	**WPA+802.1x**	**802.11i**
Clés	statique	dynamique	dynamique	dynamique	dynamique
Sécurité	faible	moyenne	moyenne	convenable	meilleure
Observation	À éviter pour une installation profession-nelle	Amélioration de la sécurité due au renouvellement des clés	Recommandé pour les réseaux domestiques	Configuration professionnelle recommandée pour une sécurité convenable	Amélioration du chiffrement avec AEP, nécessite de nouveaux équipements

C'est donc sur cette base que nous proposons l'architecture de solution de sécurité qui suit.

3. Méthodologie

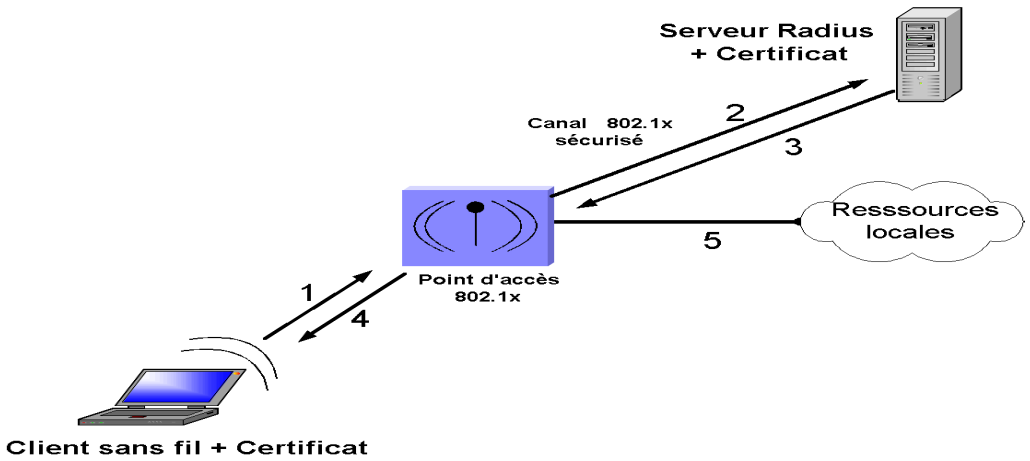

Figure 1 : *Une vue générale de la solution*

3-1. Description des étapes du processus

1 - Lorsque le client demande à accéder au réseau après avoir obtenu du serveur DHCP une adresse IP, il transmet ses informations d'identité au point d'accès sans fil. Pendant cette phase, le client ne peut avoir accès aux ressources locales ;
2 - Le point d'accès sans fil renvoie ces informations au serveur Radius. Le serveur Radius vérifie les informations d'identité, consulte sa stratégie d'accès et autorise ou refuse l'accès au client ;
3 - S'il est reconnu, le client est autorisé à accéder au réseau et échange les clés de cryptage avec le point d'accès sans fil. En fait, les clés sont générées par le serveur Radius et transmises au point d'accès sans fil via le canal sécurisé (802.1x). Si le client n'est pas reconnu par le serveur Radius, il n'est pas autorisé à accéder au réseau et la communication s'interrompt ;
4 - Grâce aux clés de cryptage, le client et le point d'accès sans fil établissent une connexion sans fil sécurisée, ce qui permet au client et au réseau interne de communiquer ;
5 - Le client commence à communiquer avec des périphériques du réseau interne.

Cette architecture implique une authentification du point d'accès au niveau du serveur, l'utilisation de certificat aussi bien par le client sans fil que par le serveur Radius. Ces certificats peuvent être générés par une Autorité de certification tierce ou par un équipement du réseau configuré pour ce fait. Nous avons préféré que ce soit le serveur Radius qui se charge de cette tâche.

3-2. Avantages et inconvénients

Cette solution permet, via l'authentification du point d'accès par le serveur Radius, d'éviter une usurpation de celui-ci par des personnes malveillantes.
Ensuite cette solution adopte EAP-TLS comme méthode d'authentification qui est utilisable avec :
- Clients : Cisco, Funk, Meetinghouse, Microsoft, Open1x (open source) [16]
- Plate-formes : Linux, Mac OS X, Windows 95/98/ME, Windows NT/2000/XP
- Serveurs Radius : Cisco, Funk, HP, FreeRADIUS (open source) [17], Meetinghouse, Microsoft (IAS).

Aussi, grâce aux certificats (générés au niveau du serveur radius), le serveur sait quels sont les clients autorisés à se connecter au réseau et quelles sont les ressources auxquelles ceux-ci ont droit. Il devient donc difficile pour un intrus de se faire passer pour un client.

En effet même si celui-ci écoute le trafic entre le client et le point d'accès, il lui sera difficile d'identifier les données échangées puisqu'elles sont encapsulées par EAP et transmises au serveur par l'intermédiaire d'un canal sécurisé. De même, le protocole de gestion des clés dynamiques utilisé (TKIP) permet d'éviter que les clés ne soient découvertes.

Il faut cependant relever que la méthode d'authentification EAP-TLS reste relativement lourde à mettre en place au niveau de la gestion des utilisateurs et des certificats et que le fait que TKIP soit basé sur RC4 pose problème.

4. Résultats et discussion

4-1. Implémentation de la solution

Recommandations :

La mise en place de cette solution de sécurité sous-entend que les opérations suivantes ont été suivies à savoir :

- Activation du cryptage WEP (au niveau des équipements) ;
- Activation du filtrage d'adresse MAC ;
- Désactivation de la diffusion du SSID (SSID Broadcast), au niveau du point d'accès ;
- Modification des mots de passe par défaut des comptes administrateurs ;
- Utilisation de "Passphrase" de taille assez grande;
- Réduction du champ d'action du point d'accès (zone de couverture).

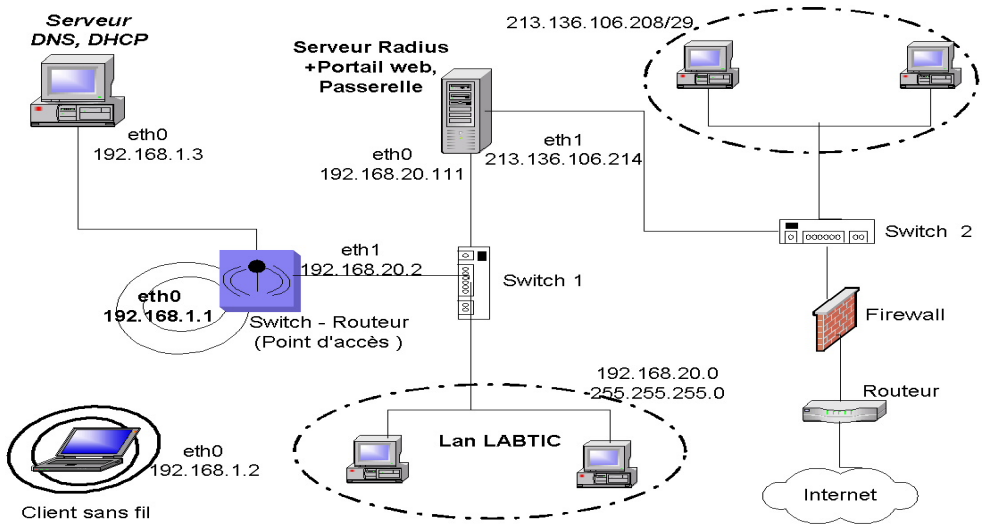

Figure 2 : *Plate forme d'expérimentation*

4-2. Plate forme de développement et technologies de mise en œuvre

Plate-forme de développement :

Plate-forme de test sous Linux (Debian sarge)
Logiciels et technologies utilisés :

- NOCAT (NoCatAuth-0.82) [8] : portail web
- RADIUS (Freeradius) [13] : serveur d'authentification
- Wpa-supplicant [18] : client WPA sous Linux

Matériel	Fabricant	Caractéristiques	Performances
Switch-Routeur	LINKSYS	Nom: BEFW11S4 V4 *Norme: 802.11b *Débit: 11 Mbps, BP :2.4 Ghz *Firware: V 1.50.14 *Type de modulation: DSSS	*Cryptage WEP 64 et 128 Bits *IPSec, filtrage MAC
Carte PCI	NETGEAR	Nom : MA311 *Norme: 802.11b *Débit Max: 11 Mbps *Type de modulation: DSSS	*Cryptage WEP 64 et 128 Bits *Cryptage WPA (NON)

4-3. Mise en oeuvre de la solution

Il faut avant tout configurer les serveurs DNS et DHCP.

4-3-1.Configuration côté serveur

4-3-1-a. Configuration du portail web (NoCatAuth-0.82)

Nous avons choisi pour des raisons économiques et d'expérimentation, d'installer le portail web et le serveur Radius sur la même machine.
Eléments nécessaires :
- Apache compilé avec mod_ssl [19] ;
- Iptables [20] pour définir les règles de filtrage ;
- Perl [21] dernière version (NoCatAuth est écrit en perl).

Après décompression (de préférence dans */usr/local/),* créer un répertoire qui contiendra les fichiers de configuration de la passerelle et du module d'authentification, puis installer et compiler les différents modules :

Génération et insertion du module gateway et insertion dans le répertoire /usr/local/nocat/gateway

[...NoCatAuth-0.82]#*make PREFIX = /usr/local/nocat/gateway gateway*

Génération du module authentification et insertion dans le répertoire /usr/local/nocat/

[...NoCatAuth-0.82]#*make PREFIX = /usr/local/nocat/ authserv*

Génération des clés PGP pour le cryptage des données

[...NoCatAuth-0.82]#*make PREFIX = /usr/local/nocat/authserv pgpkey*

Copie de la clé publique vers le répertoire contenant les modules d'authentification et attribution au démon exécutant le serveur web tous les droits sur ce répertoire, afin qu'il puisse y accéder et lire les informations. Nous avons choisi une clé de 1024 bits de longueur qui n'expire pas (uniquement pour l'expérimentation).

[...NoCatAuth-0.82]#*cp /usr/local/nocat/trustedkeys.gpg /usr/local/nocat/gateway/pgp*

[...NoCatAuth-0.82]#*chown -R www-data:www-data /usr/local/nocat/gateway/pgp*

ou ***[...NoCatAuth-0.82]#****chown -R apache:apache /usr/local/nocat/gateway/pgp*

Modification des lignes suivantes du fichier */usr/local/nocat/nocat.conf* afin d'indiquer où se situent les clés PGP, qui est la passerelle, quelle est l'adresse du réseau et les fenêtres à afficher pour l'authentification.

PGPKeyPath */usr/local/nocat/pgp*

Document Root */usr/local/nocat/htdocs*

LocalGateway *192.168.20.111*

LocalNetwork *192.168.20.0/24*

LoginForm *login.html*

LoginOKForm *login_ok.html*

Configuration de la passerelle

Fichier : */usr/local/nocat/gateway/nocat.conf*

```
GatewayMode captive
GatewayLog /usr/local/nocat/gateway/nocat.log
LoginTimeout 600
DocumentRoot /usr/local/nocat/gateway/htdocs
Owners labtic                    # utilisateur du système chargé
                                   d'administrer la passerelle
AuthServiceAddr 213.136.106.214  # adresse ip du serveur
                                   d'authentification
ExternalDevice eth1              # interface publique
InternalDevice eth0              # interface privée
LocalNetwork 213.136.106.208/29
DNSAddr 192.168.1.3              # IP du serveur DNS
IncludePorts 80 443              # ports TCP autorisés : HTTP,
                                   HTTPS
```

Configuration du serveur web Apache pour une utilisation de SSL [22]

Cela nécessite : mod_ssl [19] dernière version, openssl [25], un compilateur C ou GNU gcc. Nous utilisons pour nos tests, la version 2.0 d'apache. Pour compiler apache avec le module ssl, il convient d'ajouter au fichier de configuration */etc/httpd/conf/httpd.conf* la directive « *Include conf.d/ssl.conf* ». Ensuite il convient, après avoir configuré les fichiers *httpd.conf* et *ssl.conf*, de copier le fichier */usr/local/NoCatAuth-0.82/authserv.conf* dans le répertoire */usr/local/nocat/etc/* (le fichier *ssl.conf* fait un lien vers ce fichier).

Création des comptes

Le compte administrateur est celui du « owners » que nous avons indiqué dans le fichier */usr/local/nocat/gateway/nocat.conf*

Il existe un outil convivial [23] en mode graphique qui permet d'administrer le serveur. Il nécessite l'installation de Webmin [24].

[...NoCatAuth-0.82]#*/usr/local/nocat/bin/admintool -c labtic labtic*
(création du compte labtic avec mot de passe labtic)

[...NoCatAuth-0.82]#*/usr/local/nocat/bin/admintool -A labtic labtic*
(Attribution du droit d'administration)

Exécution de Nocat

La configuration achevée, on passe à l'exécution de nos serveurs web, DNS, et Nocat.

[...NoCatAuth-0.82]#*/etc/rc.d/init.d/named restart*

[...NoCatAuth-0.82]#*apachectl startssl*

[...NoCatAuth-0.82]#*/usr/local/nocat/gateway/gateway*

4-3-1-b. Configuration de freeradius (version 1.0.1) [17]

Nous allons enfin configurer notre serveur pour une authentification basée sur le protocole EAP/TLS (supporté par nos équipements). La description de ce protocole de transport est largement décrite dans la RFC 2716 [6].
Fichier à configurer : *clients.conf, radiusd.conf, user*

Configuration de *clients.conf* (*/usr/local/radius/etc/raddb/clients.conf*). Ce fichier définit tous les points d'accès autorisés à ouvrir une session radius sur le serveur.

```
client 192.168.20.2 {
      secret  = secpartlabtic   # mot de passe pour la communication entre
                                # serveur et l' AP
      shortname =  linksys      # identifiant du réseau (SSID)
      nastype = other           # type d'AP (cisco, ...)
}
```

Configuration de *radiusd.com* (*/usr/local/radius/etc/raddb/radiusd.conf*) pour l'utilisation de EAP-TLS.

```
User = labtic
Group = labtic
modules {
        eap {
             default_eap_type = tls
             timer_expire  = 60
             tls {
        private_key_password = secpartlabtic
        private_key_file = /usl/local/radius/etc/1x/cert-srv.pem
        certificate_file = /usr/local/radius/etc/1x/cert-srv.pem
        CA_file = /usr/local/radius/etc/1x/root.pem
        dh_file = /usr/local/radius/etc/1x/dh
        random_file = /usr/local/radius/etc/1x/random
        fragment_size = 1024
        include_length = yes
         }
          }
authorize {
            …
              eap           # autorisation de l'authentification EAP files
        }
authenticate {
                eap         # permission de l'authentification EAP
           }
```

Configuration de user *(/usr/local/radius/etc/raddb/user).* Celui-ci contient les clients autorisés à se connecter et à être validés par le serveur. Nous l'utilisons dans le cadre de cet article mais il faut savoir qu'il est possible d'utiliser freeradius avec postgresql, mysql, LDAP afin de sauvegarder les données concernant les clients.

"labtic-wireless" Auth-Type:= EAP
"labtic-visitor" User-password== "labticvisitor"

NB : Il faut noter que les certificats sont générés grâce à Openssl [25]. Il faut donc l'installer et le configurer [26]. Pour être certain que l'authentification se fera sans problème, il est impératif de vérifier que tous les certificats générés ont été copiés dans les bons fichiers :

- Au niveau du serveur :

/usl/local/radius/etc/1x/cert-srv.pem , contenant la clé privée du serveur
/usr/local/radius/etc/1x/cert-srv.pem , contient le certificat
/usr/local/radius/etc/1x/root.pem , contient le certificat d'autorité (serveur radius)

- Chez les clients :

/etc/cert/ca.pem , contient le certificat d'autorité (serveur radius)
/etc/cert/user.pem
/etc/cert/user.prv , contient la clé privée

4-3-1-c. Configuration du point d'accès

La configuration du point d'accès nécessite la mise à jour du firmware (pour la prise en compte du 802.1x et WPA). Ensuite la déclaration de deux types de SSID à savoir :
Un pour les clients sans fil du LABTIC (labtic-wireless) et l'autre pour les visiteurs (labtic-visitor).

4-3-2. Configuration des clients sans fil du LABTIC

Pour les clients, nous utilisons wpa-supplicant [18].
Après installation, le fichier de configuration se trouve dans */etc/wpa-supplicant*.
Nous le modifions pour l'adapter à notre configuration.

```
ctrl_interface = /var/run/wpa_supplicant
ctrl_interface_group=0
eapol_version=1
ap_scan=1
network={
        ssid="labtic_wireless"
        proto=RSN
        key_mgmt=WPA-EAP
        pairwise=TKIP
        group=TKIP
        EAP=TLS
        ca_cert="/etc/cert/ca.pem"
        client_cert=/etc/cert/user.pem
        private_key="/etc/cert/user.pem"
        priority=1
        }
```

Quand aux visiteurs, il n'existe pas de configuration particulière. En effet ceux-ci ne sont pas obligés d'utiliser des certificats. Il suffit donc de les configurer pour utiliser un serveur DHCP, d'indiquer l'adresse IP de ce serveur ***(ici 192.168.1.3)*** et d'indiquer comme SSID : *labtic-visitor*.

4-3-3. Mise en marche et test

Mise en marche :

- Pour lancer le serveur radius : radius-X-A & Résultat de l'exécution si tout est correct :

```
Starting_reading configuration files
reread_config :  reading radiusd.conf
Config :   including file : /usr/local/etc/raddb/proxy.conf
Config :   including file : /usr/local/etc/raddb/clients.conf
Config :   including file : /usr/local/etc/raddb/snmp.conf
Config :   including file : /usr/local/etc/raddb/eap.conf
Config :   including file : /usr/local/etc/raddb/sql.conf
.....
Module : Loaded eap
.....
rlm_eap : Loaded and initialized type tls
.....
Listening on authentication *: 1812
Listening on accounting *: 1813
Ready to process requests
```

- Au niveau du client : *wpa-supplicant -i wlan0 -c /etc/wpa-supplicant.conf –B*

Tests

Pour tester, il suffit de configurer la carte réseau PCI du client en indiquant l'usage de DHCP, l'adresse du serveur DHCP, labtic-visitor ou labtic-wireless comme SSID en fonction du type de client puis de lancer un navigateur. Si tout est correct, la fenêtre de Nocat apparaît, demandant pour la première connexion une authentification. On saisit l'un des comptes qu'on a créé pour se connecter.

Remarque: si un visiteur choisit comme SSID labtic-wireless, celui-ci ne pourra pas s'authentifier dans la mesure où il ne dispose pas de certificat délivré par le serveur Radius.

5. Conclusion

L'émergence des technologies sans fil dans le monde des réseaux informatiques a entraîné l'apparition de problématiques tout à fait nouvelles. Les entreprises dotées d'un système sans fil sont particulièrement confrontées à des sérieux problèmes de sécurité.

Nous avons présenté dans ce article ce que pourrait être une architecture de solution sécurité dans un réseau sans fil pour une entreprise soucieuse des risques liés aux « sans fil ». Cette architecture comme nous avons eu à le dire permet d'assurer un certain niveau de sécurité en protégeant d'une part l'accès au réseau par des mécanismes d'authentification, d'autorisation et de gestion des comptes, et d'autre part en sécurisant les communications grâce à différents protocoles de cryptage utilisés.

Cette solution, en attendant la vulgarisation d'équipement capable d'implémenter la nouvelle norme en matière de sécurité (802.11i), reste tout à fait acceptable voire nécessaire.

Références

[1] - Ratifié en septembre 1999 en tant que IEEE Std. 802.11b , http://standards.ieee.org/reading/ieee/interp/802.11b-1999.html

[2] - Protocole de cryptage utilisant l'algorithme RC4 et basé sur des clés de 64, 128 bits

[3] - FLUHRER, MANTIN et SHAMIR, "*Weaknesses in the key scheduling algorithm of RC4*", English Annual Workshop on Selected Areas in Cryptography (08/2001). Editeur: Springer-Verlag London, UK.

[4] - STUBBLEFIELD, IOANNIDIS et RUBIN, "*Using the Fuhrer, Mantin and Shamir Attack to Break WEP*", AT&T Labs Technical Report, TD-4ZCPZZ (08/2001).

[5] - RFC 3580 définissant le standard 802.1x , http://www.ieee802.org/1/pages/802.1x.html

[6] - RFC 2716 consacrée à EAP-TLS, http://www.faqs.org/rfcs/rfc2716.html

[7] - Site official de la Wi-Fi Alliance organe qui à propose le WPA, http://www.wi-fi.org/

[8] - Site officiel de NOCAT, http://nocat.net/

[9] - IEEE 802.11i draft & Call for IEEE 802.11i draft & Call For Interest on Link Security for IEEE Interest on Link Security for IEEE 802 Networks, http://grouper.ieee.org/groups/802/3/efm/public/nov02/sec/halasz_sec_1_1102.pdf

[10] - Outil d'espionage Airsnort, Wepcrack, http://airsnort.sourceforce.net , http:/wepcrack.sourceforce.net
[11] - RFC 2284 PPP Extensible Authentication Protocol (EAP), http://www.faqs.org/rfcs/rfc2284.html
[12] - Site officiel de l'IETF , http://www.ietf.org
[13] - Remote Authentication Dial In User Service (RADIUS), http://www.faqs.org/rfcs/rfc2865.html
[14] - Démonstration de l'université de Maryland, http://www.cs.umd.edu/%7Ewaa/1x.pdf
[15] - RFC 3962 Advanced Encryption Standard (AES) Encryption for Kerberos 5, http://www.faqs.org/rfcs/rfc3962.html
[16] - Open source implementation of IEEE 802.1x, http:// www.open1x.org
[17] - Site official de Freeradius, http://www.freeradius.org
[18] - Logiciel wpa-supplicant, http://hostap.epitest.fi/wpa_supplicant/
[19] - Module permettant au serveur apache de supporter SSL (utiliser pour protéger les transactions), http://httpd.apache.org/docs/2.0/mod/mod_ssl.html
[20] - Site officiel du Netfilter/iptables , http://www.netfilter.org/
[21] - documentation sur perl , http://www.perl.com/pub/q/documentation/
[22] - SSL (Secure Socket Layer), http://httpd.apache.org/docs/2.0/ssl/ssl_intro.html
[23] - Outil graphique permettant d'administrer NoCat, http://ovh.dl.sourceforge.net/sourceforge/nocat-webmin/nocat-051.wbm
[24] - Site officiel de webmin , http://www.webmin.com
[25] - Site officiel consacré à OpenSSL, http://www.openssl.org
[26] - Création de certificat avec OpenSSL , http://www.tldp.org/HOWTO/SSL-Certificates-HOWTO/ .