IGC et certificats pour sécuriser les accès des utilisateurs nomades

François Morris
LMCP
Université Pierre et Marie Curie, campus Boucicaut, case 115, 75252 PARIS Cedex 05
Francois.Morris@lmcp.jussieu.fr
date: 8 octobre 2003

Résumé

La demande croissante d'accès de la part d'utilisateurs nomades, le remplacement des postes fixes par des ordinateurs portables, le développement des réseaux sans fil (Wi-Fi), la nécessité de fournir un accès Internet aux visiteurs posent de nouveaux et sérieux risques en matière de sécurité mais constituent aussi un intéressant défi à relever. La solution passe par l'utilisation de protocoles sécurisés à l'aide de techniques cryptographiques. Le coût de mise en œuvre de la gestion des clés a souvent été un frein à l'utilisation de ces techniques. Le déploiement d'un IGC au CNRS avec tous les coûts que cela induit étant désormais acquis la situation a changé. Il devient désormais possible d'utiliser des certificats pour sécuriser les accès à un coût devenu raisonnable. Différentes solutions sont disponibles : version sécurisée des protocoles classiques (HTTPS, SMTPS, IMAPS, POP3S, portail captif, VPN et tunnel IPSec, authentification au niveau 2 (802.1x) Nous les passerons en revue et effectuerons un bilan de leurs avantages et inconvénients.

Mots clés

Authentification, IGC, nomades

1 Introduction

Nous travaillons dans un environnement où beaucoup d'utilisateurs sont des nomades. Ils souhaitent légitimement être capables d'accéder au système d'information depuis n'importe où dans le monde et non seulement de leur bureau. Cela peut être le lieu où ils font leur enseignement, la salle où se trouve leur expérience, leur domicile, un autre laboratoire, un palais des congrès, un cybercafé, un hôtel, un « hot spot » dans un aéroport. La tendance actuelle est au remplacement des stations de travail fixes par des ordinateurs portables. En outre, il faut aussi pouvoir offrir aux visiteurs un accès à Internet. Les moyens de connexion au réseau sont variés : Ethernet, modem, ADSL, Wi-Fi.
Pour le responsable de la sécurité des systèmes d'information, c'est un vrai cauchemar mais aussi un défi très intéressant à relever. Voici un petit inventaire des problèmes rencontrés. Un ordinateur portable peut avoir récupérer un virus, un ver ou un cheval de Troie alors qu'il était connecté à l'extérieur et est susceptible de le transmettre lorsqu'il est de nouveau connecté sur le réseau interne. Dans un environnement universitaire il est impossible d'avoir un contrôle strict sur les personnes accédant aux bâtiments et par conséquent un individu arpentant les locaux et ayant des visées hostiles peut aisément trouver une prise Ethernet et se connecter au réseau interne. Un réseau sans fil (Wi-Fi) peut déborder sur la voie publique et être accessible par n'importe qui. L'utilisation pour l'authentification du couple identifiant, mot de passe circulant en clair sur le réseau qui est de fait la règle générale n'est pas suffisamment robuste car à la merci d'un pirate écoutant sur le réseau.
Un contrôle périmétrique à l'aide de coupe-feu est insuffisant lorsque les frontières du réseau s'étendent au monde entier. Comment savoir qui est responsable de telle ou telle action particulièrement dommageable quand il n'y a plus de correspondance fixe entre les machines, les utilisateurs, les adresses, les locaux.
Les techniques cryptographiques permettent d'apporter des solutions à ces problèmes. Quand on parle de cryptographie on pense généralement confidentialité. Mais on oublie souvent qu'elle peut aussi être utilisée pour l'authentification et le contrôle de l'intégrité des informations. Dans bien des cas d'ailleurs, il est plus important de savoir que l'on discute avec le bon interlocuteur que de chiffrer le message pour éviter qu'il ne soit intercepté.
.

2 IGC

La vraie difficulté avec les techniques cryptographiques réside dans l'échange et la gestion des clés de chiffrement. La cryptographie à clés asymétriques [1] [2] a apporté une solution élégante à ce problème. Une infrastructure de gestion de clés (IGC) permet sa réalisation. Cependant le coût de sa mise en œuvre, d'ailleurs plus en termes organisationnels que techniques, a souvent fait hésiter à l'utiliser. Maintenant que le déploiement de l'IGC du CNRS [3] est acquis, l'utilisation de certificats pour sécuriser les accès des nomades devient tout à fait envisageable. En effet les procédures pour délivrer les certificats sont opérationnelles. Les utilisateurs qui ne possèdent pas encore un certificat doivent en demander un et

l'installer sur leur machine. Cette opération s'est avérée ne pas poser de difficultés particulières. Il faut aussi installer des certificats sur les serveurs. Cette dernière opération ne concerne pas les utilisateurs mais uniquement ceux qui sont en charge de l'administration du système.
L'utilisation de certificat personnel est une façon élégante de réaliser une authentification unique (« Single Sign On » ou SSO). Lorsque l'on connaît les difficultés et les conséquences en terme de sécurité posées aux utilisateurs par la multiplication des mots de passe ce n'est pas un mince avantage. Le fait que le secret utilisé pour l'authentification à la différence des mots de passe classiques ne réside jamais sur le serveur est une excellente chose pour la tranquillité d'un administrateur système.

3 Webmail, webftp

Un « webmail » est une application qui tourne sur un serveur Web. Elle permet à un utilisateur à travers des pages HTML générées dynamiquement d'accéder à sa messagerie électronique pour lire ses messages, les supprimer, les archiver, en envoyer de nouveau. Comme le seul préalable est que l'utilisateur dispose d'un simple navigateur, pratiquement n'importe quelle machine dans le monde peut être utilisée à cette fin. Il existe de nombreuses applications « webmail », parmi elles : SquirrelMail [4], JAWMAIL [5], Horde IMP [6] qui est celle que nous utilisons.
Les échanges avec le serveur sont chiffrés en utilisant le protocole HTTPS qui est la variante sécurisée de HTTP. L'authentification de l'utilisateur est effectuée à l'aide d'un classique couple identifiant, mot de passe en clair. Puisque tout le trafic est chiffré, ce mot de passe ne peut être intercepté sur le réseau. L'authentification n'est certes pas aussi forte que celle que l'on pourrait avoir avec un certificat personnel. Mais ce serait une très mauvaise idée en matière de sécurité que d'installer son certificat et surtout la clé privée associée sur une machine que l'on ne contrôle pas. En effet la clé privée pourrait après coup être récupérée sur le disque de l'ordinateur et même si celle-ci est chiffrée en utilisant un mot de passe rien ne garantit que ce dernier ne va pas être cassé à l'aide d'une attaque par force brute ou par dictionnaire.
Un autre risque à considérer est celui des attaques de type entremetteur[1] où le pirate déroute la connexion vers sa propre machine plutôt que vers le serveur légitime. Le pirate peut alors présenter une fausse page d'identification et récupérer le mot de passe. Certes le navigateur est censé vérifier le certificat du serveur mais pratiquement l'utilisateur a toujours la possibilité de passer outre. Cela permet d'utiliser pour les serveurs des certificats autosignés ou bien signés par des autorités de certification qui ne sont pas installés par défaut dans le navigateur comme c'est le cas pour celui du CNRS. Peut-on par ailleurs faire totalement confiance aux autorités installées par défaut ? La bonne démarche pour un utilisateur serait d'installer à partir du réseau sur la machine le certificat racine du CNRS en vérifiant que l'empreinte fournie correspond bien à celle que l'on a récupéré par un canal sûr et que l'on conserve sur une feuille de papier rangé dans son porte-feuille par exemple. Si le certificat est déjà installé, il faudrait en vérifier l'empreinte. Disons le tout de suite ce n'est absolument pas réaliste.
La qualité de la confidentialité et de la protection du mot de passe sont celles du protocole SSL. Sur une machine que vous ne contrôlez pas, il existe toujours des risques. Un logiciel espion a pu être installé, les touches utilisées pour entrer le mot de passe, les messages affichés peuvent être interceptés.
Cependant la sécurité doit être considérée comme assez bonne. Même si ce n'est pas parfait, un webmail est la meilleure méthode pour accéder à son courrier électronique depuis une machine non contrôlée comme cela peur-être le cas dans un cyber-café.
L'expérience a montré que nombre d'utilisateurs trouvent l'utilisation d'un webmail si commode que même sur le réseau local il le préfère à des produits spécifiques comme Outlook, Messenger. Il faut donc prendre garde lors de l'installation d'un service webmail à le dimensionner correctement en tenant aussi compte des connexions depuis l'intérieur. Dans ce contexte les problèmes de sécurité sont bien moindres que ceux signalés précédemment, le certificat racine nécessaire pour valider celui du serveur est généralement installé ou peut l'être facilement.
Il faut cependant prévenir les gens qu'ils ne doivent en aucun cas utiliser n'importe quel webmail public. En effet on trouve un certain nombre de sites offrant un service de webmail, il suffit de donner le nom de son serveur IMAP pour pouvoir consulter son courrier. Ceci est très dangereux car on fournit son identifiant et son mot de passe sans aucune garantie.
Pour l'administrateur l'installation d'un webmail est loin d'être trivial. Il faut un serveur web supportant le protocole HTTPS, la gestion des pages dynamiques (généralement PHP). Il faut aussi un serveur IMAP pour gérer le courrier électronique. Très souvent il faut aussi une base de donnée comme MySQL pour stocker la configuration du produit. Mais l'effort en vaut vraiment la peine. Le service webmail est plébiscité par les utilisateurs et permet de répondre à la demande principale des utilisateurs nomades qui est de consulter ses messages. C'est incontestablement le premier service à fournir pour les nomades.
Une autre application analogue dans tous les aspects au webmail est le webFTP. Ce service utilise FTP pour accéder aux fichiers de l'utilisateur. Parmi les produits disponibles on peut citer celui développé par Albert Shih [7] ou bien Horde Gollem [8] qui est que nous utilisons. Transférer ses fichiers à partir d'une machine d'un cyber-café n'est sûrement pas très

[1] Parfois appelé en anglais « Man in the Middle ».

avisé mais ce n'est probablement pas très utile non plus. Par contre si vous vous trouvez dans le bureau d'un collègue cela peut être très commode pour récupérer le fichier que l'on a oublié. Dans ce cas les risques restent très limités car l'on peut raisonnablement considérer que la machine est relativement sûre.

4 Protocoles sécurisés pour le courrier électronique

Dans une architecture aujourd'hui habituelle le courrier électronique arrive sur un serveur et est stocké sur celui-ci. Les utilisateurs récupèrent leurs messages en utilisant une application qui se connecte au serveur en utilisant le protocole POP3 ou IMAP. IMAP offre plus de fonctionnalités que POP3. Pour simplifier POP3 permet uniquement de récupérer d'un coup tous les messages et IMAP permet de décider ce que l'on fait pour chaque message comme le lire, le détruire sans le transférer, l'archiver. L'inconvénient est que ces protocoles n'offrent aucune sécurité. Non seulement il n'y a aucune confidentialité car les messages circulent en clair sur le réseau mais surtout l'authentification fait appel à un simple couple identifiant, mot de passe. Le mot de passe transite en clair et est donc susceptible d'être intercepté et ce d'autant plus que les clients se connectent à intervalles réguliers pour voir si de nouveaux messages ne sont pas arrivés. Certes il existe quelques variantes des protocoles qui permettent d'utiliser des méthodes plus sûres d'authentification comme un défi, réponse pour vérifier le mot de passe. Il reste qu'elles ne sont pas implémentées partout et qu'assurer l'interopérabilité entre les différents produits est quasi impossible. La conclusion est qu'il ne faut jamais permettre de consulter ses messages à travers Internet depuis l'extérieur en utilisant les protocoles POP3 ou IMAP. Une possibilité est de les filtrer au niveau du routeur d'entrée ou du coupe-feu.

SSL (Secure Socket Layer) est le protocole développé à l'origine par Netscape [9] pour sécuriser les connexions et notamment celles aux serveurs web (HTTPS). TLS (Transport Layer Security) est la version 3 de SSL repris par l'IETF [10]. Les données échangées à l'aide de ce protocole sont chiffrées ce qui en assure la confidentialité. Afin de s'authentifier auprès du client le serveur envoie son propre certificat. A l'aide du certificat de l'autorité de certification qui a délivré le certificat du serveur le client vérifie que le certificat reçu du serveur est bien correct et qu'il est donc connecté au bon serveur et non pas à un pirate. Réciproquement le serveur a la possibilité de demander au client son certificat afin de pouvoir l'authentifier avec certitude. Cette dernière possibilité n'est pas toujours utilisée.

Il existe des versions sécurisées à l'aide de SSL/TLS des protocoles d'accès au courrier : POP3S et IMAPS. Avec POP3S et IMAPS les échanges sont chiffrés, ce qui d'une part assure la confidentialité mais surtout évite de faire circuler en clair le mot de passe. On élimine ainsi une des causes majeures d'intrusion sur les réseaux. Les versions récentes des outils de messagerie comme Outlook ou Netscape Messenger déterminent si le serveur de courrier supporte le protocole sécurisé (TLS) et dans ce cas bascule automatiquement en mode sécurisé en lançant une commande « startls ». Ceci est très appréciable car l'utilisateur n'a rien à faire pour obtenir par défaut une connexion sécurisée. L'administrateur a juste à installer une version supportant TLS du serveur et à le configurer pour accepter les connexions sécurisées et refuser celles qui ne le sont pas du moins pour celles en provenance de l'extérieur.

Il est possible avec les versions sécurisées des protocoles d'utiliser des certificats personnels pour authentifier l'utilisateur. La méthode d'authentification est alors évidemment plus forte. Elle permet aussi d'utiliser des supports matériels (carte à puce, jeton USB). Puisque l'utilisateur est parfaitement authentifié à l'aide de son certificat il n'est théoriquement plus nécessaire d'utiliser aussi une authentification par mot de passe. Pour cela le serveur IMAP envoie la directive « preauth » qui signifie au client qu'il est déjà authentifié et qu'il n'a pas besoin d 'envoyer son identité. Lors de nos derniers essais seuls Mozilla et Netscape permettaient d'utiliser des certificats clients. Outlook ne permet que de chiffrer. Pour permettre la pré-authentification il a fallu modifier légèrement le serveur IMAP (Cyrus [11])[2].

L'envoi de messages se fait en se connectant à un serveur à l'aide du protocole SMTP. Ce serveur va relayer le message vers d'autres serveurs afin qu'il soit distribué au bon destinataire. Pour éviter de servir d'amplification de courrier non sollicités (SPAM) une bonne pratique est d'interdire sur tout serveur SMTP le relais de messages dont les adresses des expéditeur et destinataire sont toutes deux externes. Seuls sont autorisés les messages émis à l'intérieur pour l'extérieur et ceux provenant de l'extérieur vers l'intérieur. Ceci oblige un utilisateur nomade connectant son ordinateur portable sur un réseau à se renseigner pour connaître le relais SMTP local et à modifier en conséquence la configuration de l'outil de messagerie puisque le relais qu'il utilisait lorsqu'il était à l'intérieur va refuser ses messages. Pour éviter ceci il est possible d'utiliser SMTPS la version sécurisée du protocole avec authentification par certificat personnel. En effet autoriser le serveur SMTP à relayer des messages provenant de l'extérieur mais d'utilisateurs dûment authentifiés n'ouvre aucune brèche dans la sécurité. En outre les échanges sont chiffrés mais cela n'apporte pas grand chose en matière de confidentialité puisque lorsque le message transite de relais en relais seule la première étape est chiffrée, les autres ne l'étant généralement pas. Evidemment l'administrateur doit installer un serveur SMTP supportant l'authentification du client par certificat et le configurer en conséquence.

[2] Sans modification l'utilisateur doit spécifier son identifiant et son mot de passe ce qui est inutile puisqu'il est déjà authentifié par son certificat. Il reste toujours la possibilité de faire mémoriser le mot de passe par l'outil de messagerie ce qui ne pose pas de problèmes de sécurité moyennant certaines précautions (mot de passe réservé au courrier ou rangé dans la zone protégée un autre mot de passe et qui contient aussi la clé privée du certificat).

Notre expérience a montré qu'en utilisant les versions sécurisées des protocoles IMAP et SMTP avec Mozilla ou Netscape Messenger comme client de messagerie, une version légèrement modifiée de Cyrus comme serveur IMAP et une version de Postfix supportant TLS comme relais SMTP nous parvenions à une situation très confortable pour les utilisateurs nomades. En effet ceux-ci n'ont jamais à changer la configuration de leur ordinateur portable. En outre l'authentification par certificat se révèle plus commode car il suffit de donner un fois pour toute au début son mot de passe pour déverrouiller sa clé privée. Ceci sera valable pour toutes la applications utilisant une authentification par certificat : réception de messages, envoi de messages, connexion à un Intranet (cf. infra). Evidemment le chiffrement a aussi lieu lorsque l'on se trouve sur le réseau interne ce qui n'est probablement pas nécessaire en matière de sécurité mais reste tout à fait acceptable en termes de performances.
Pour les ordinateurs portables et peut-être aussi les ordinateurs personnels à domicile nous estimons que la façon la plus efficace et la plus simple d'utilisation pour gérer sa messagerie est d'utiliser les protocoles sécurisés IMAPS et SMTPS avec authentification du client par certificat personnel. Le vrai problème reste la disponibilité des outils de messagerie pour l'utilisateur. Il est toujours possible de changer le logiciel d'un serveur même si n'est pas toujours très simple. Par contre changer les habitudes d'un utilisateur est une autre histoire.

5 Intranet

5.1 HTTPS

Un certain nombre d'informations à usage interne publiées sur un serveur Web n'ont pas à être visibles de l'extérieur. Il est relativement facile de restreindre l'accès uniquement aux machines connectées sur le réseau interne. Il est cependant souhaitable qu'un utilisateur nomade puisse aussi accéder aux informations depuis l'extérieur dans des conditions de sécurité satisfaisante. La solution passe généralement par l'utilisation du protocole HTTPS. Comme déjà vu (cf. supra) HTTPS assure la confidentialité des échanges de données. Il permet aussi théoriquement au client de savoir qu'il s'adresse au bon serveur sachant que pour permettre un fonctionnement dans des situations où les différents certificats des autorités de certification ne sont pas correctement installés les navigateurs permettent généralement de passer outre à une vérification infructueuse du certificat du serveur.

5.2 Authentification du client

Généralement le serveur web a besoin d'authentifier de façon fiable le client. Mais ce n'est pas toujours le cas, un serveur de commerce électronique pourra se contenter de demander le numéro de la carte bancaire. Evidemment ce numéro devra être chiffré lors du transfert ce qu'assure le protocole HTTPS. Cette exigence provenant d'ailleurs plus du client qui ne souhaite pas divulguer son numéro de carte au monde entier que du commerçant..

5.2.1 Identifiant – mot de passe

Différents mécanismes d'authentification sont prévus par le protocole HTTP. Pratiquement seul le mode « basique » qui repose sur un identifiant et un mot de passe en clair est couramment employé, les autres n'étant généralement pas implémentés sur les différents produits. Ce qui n'est pas vraiment gênant en matière de sécurité car bien évidemment l'ensemble du trafic doit être chiffré en utilisant HTTPS. Il est ainsi possible de spécifier qu'un ensemble de pages web n'est accessible qu'avec un mot de passe. Cela se réalise à l'aide de directives spécifiques soit dans le fichier de configuration générale du serveur web, soit dans un fichier associé à un nœud dans l'arborescence des pages.
Il est aussi possible à une application web de gérer son propre mécanisme d'authentification. C'est d'ailleurs ce qui est utilisé, comme vu précédemment, dans les application de type webmail

5.2.2 Certificat client

Les dernières versions de SSL et par conséquent HTTPS permettent une authentification du client à l'aide de certificats. Dans ce cas le serveur demande au client son certificat personnel et s'assure qu'il possède bien la clé privée associée au certificat. Par des directives appropriées dans les fichiers de configuration il est possible de spécifier que certaines pages ne seront accessibles qu'uniquement à une personne ayant un certificat valide et d'utiliser les informations contenues dans ce certificat comme l'émetteur ou l'identité du détenteur pour affiner encore plus l'accès.
Cette dernière méthode présente plusieurs avantages. Elle est plus forte que celle reposant sur un mot de passe. Ce qui peut encore être amélioré par l'utilisation de support physique pour le certificat comme une carte à puce ou un jeton USB. Dans le cas où il existerait déjà une IGC l'administration se révèle nettement plus simple. En effet le serveur n'a pas à gérer de mot de passe, il a simplement à s'occuper des identités des personnes qu'il veut autoriser. Pour l'utilisateur cela se révèle aussi beaucoup plus simple car il n'a qu'un seul mot de passe à connaître celui qui déverrouille sa clé privée et pratiquement il n'est demandé qu'une seul fois. C'est effectivement une méthode pour réaliser une authentification unique « Single Sign On » ou SSO.

5.3 Authentification du serveur

Pour authentifier le serveur, le client vérifie le certificat présenté à l'aide de la chaîne des certificats des autorités racine et intermédiaires. Ceci exige que le certificat de l'autorité racine soit bien installé sur la machine du client. Cette authentification du serveur est aussi extrêmement importante, si on veut éviter toute une classe d'attaques où le pirate cherche à se faire passer pour la machine légitime.

5.4 Relais HTTP inverse

Un relais HTTP inverse est un serveur qui reçoit les requêtes provenant du monde entier pour les retransmettre à un serveur web[3]. L'intérêt par rapport à une connexion directe est d'une part que ce relais peut effectuer des filtrages mais surtout il peut utiliser HTTPS, authentifier le client avant de transmettre les requête en HTTP au serveur web final. On peut donc sécuriser ce dernier, sans avoir à le modifier.

Avec le développement des services Web presque tout est encapsulé dans HTTP et peut par conséquent être sécurisé en utilisant HTTPS. Le Webmail, vu précédemment, en est un exemple.

6 TLS, SSH

Comme nous l'avons déjà vu les protocoles HTTP, IMAP, POP3, SMTP possèdent une version sécurisée. L'IETF a aussi développé des versions sécurisées par TLS de protocoles comme telnet et FTP.
Des versions sécurisées des serveurs et clients telnet et FTP ont été développées à partir des produits classiques pour Unix. Malheureusement les produits pour Windows que nous avons essayé se sont révélés difficilement utilisables. En effet chaque application cliente comme telnet ou FTP gérait ses certificats et clés indépendamment des autres applications et en particulier du navigateur Web. En outre certains produits testés demandaient à l'installation une fois pour toutes le mot de passe permettant de déverrouiller la clé privée de l'utilisateur pour la ranger dans un fichier ou le registre Windows[4]. Ceci impose de faire totalement confiance au mécanisme d'authentification et de sécurité de Windows.
La connexion à distance sur une machine ou le transfert de fichiers étant des besoins légitimes réclamés par les utilisateurs nomades, il a fallu se tourner vers l'alternative SSH. SSH est à la fois le nom d'un protocole, celui d'un produit et enfin d'une société développant le produit [12]. Il existe une implémentation « libre » du produit openssh [13]. SSH est une alternative à telnet pour les connexions à distance et à FTP pour les transferts de fichiers. Les échanges sont chiffrés, le serveur est authentifié par un mécanisme à clés asymétriques, le client est authentifié soit avec un mot de passe soit de préférence par un mécanisme de clés asymétriques. Les méthodes employées sont donc très semblables à celles utilisées par SSL/TLS. SSH utilise un système propre pour gérer ses clés publiques. Les nouvelles versions du produit SSH développé par SSH introduisent le support des certificats X509. Il existe aussi quelques développements pour introduire la gestion des certificats X509 dans openssh.

7 Portail captif

Afin de permettre de connecter des ordinateurs portables sur notre réseau sans abaisser le niveau de sécurité nous avons créé une zone particulière (VLAN) réservé à cet usage. Un portail captif [14] est une application web qui utilise les méthodes standard du protocole HTTP pour authentifier les utilisateurs (mot de passe, certificat client) et leur donner des droits. En fonctions des droits obtenus, les règles sont modifiées dynamiquement sur le coupe-feu pour l'ordinateur portable correspondant.
Après avoir étudier différentes solutions, nous avons décidé de développer notre propre application [15]. En effet les produits existant n'utilisaient pas le certificat client pour l'authentification mais des méthodes basées sur un serveur Radius ce qui complique sérieusement le code. L'authentification par mot de passe est intégrée à HTTP et celle par certificat à HTTPS et il n'y a aucun code à développer.
Nous utilisons un PC sous Linux comme passerelle entre le VLAN des ordinateurs portables et le réseau interne ou Internet. Le portail est constitué d'une application web en réalité il s'agit de quelques scripts écrits en PHP. Cette application tourne sur un serveur Apache et envoie des requêtes à un démon gérant les règles de filtrage. Ce démon est un script écrit en PERL qui génère des commandes « iptables ».
Au départ les règles de filtrages sont définies de telle sorte qu'une machine se connectant sur le VLAN des portables a uniquement le droit de faire du DHCP pour récupérer son adresse IP. Ensuite ayant récupérée son adresse IP, seules les protocoles DNS afin de pouvoir assurer la traduction des noms en adresses IP ainsi que HTTP et HTTPS sont autorisés. Toute requête HTTP ou HTTPS est automatiquement redirigée vers la page portail du serveur Apache. Une fois authentifié

[3] Par rapport à un cache Web il fonctionne dans l'autre sens d'où le nom.
[4] Il vrai que Windows ne fait guère autrement avec ses magasins de certificats logiciels.

les règles de filtrages sont établies en fonction des droits associés. Un membre dûment authentifié du laboratoire aura accès au réseau interne. Un visiteur est authentifié à l'aide d'un simple mot de passe et est seulement autorisé à accéder à Internet et peut-être une imprimante locale. Le mot de passe utilisé dans ce cas n'est pas vraiment secret, il a pour but essentiel de décourager dans des locaux où le contrôle d'accès est quasi inexistant, les maraudeurs qui trouvant une prise Ethernet profitent de l'accès à Internet. Il s'agit de se prémunir contre une consommation indue de la bande passante et d'éviter de voir sa responsabilité engagée en cas d'usage non respectueux des lois.
La récente mise sur le marché de commutateurs supportant l'authentification des stations de travail à l'aide du protocole 802.1x offre une nouvelle solution qui peut s'avérer plus simple pour authentifier les ordinateurs portables.
Ce mécanisme pourrait être envisagé pour sécuriser dans une certaine mesure les accès Wi-Fi dans le cas où l'utilisation de 802.1x se révélerait impossible.

8 IPSec

IPSec [16] est un protocole destiné à sécuriser IP. Développé à l'origine dans le cadre de IP V6 (la future version d'IP) il est désormais intégré à IP V4 (la version actuelle d'IP). IPSec permet d'authentifier les extrémités d'une connexion et offre la possibilité de chiffrer les échanges. Il permet aussi de s'assurer que les données transmises n'ont pas été altérées.
Deux modes sont disponibles : le mode transport pour sécuriser le trafic entre deux machines et le mode tunnel qui est utilisé pour construire des réseaux privés virtuels (« Virtual Private Network » ou VPN)[21].
Différentes méthodes pour la gestion et la distribution des clés sont permises par IPSec. Lorsque l'on dispose déjà d'une IGC, la méthode utilisant les certificats X509 s'impose.
IPSec fonctionne au niveau de la couche réseau, il est donc possible de sécuriser les transactions sans avoir à modifier les applications.
Maintenant plusieurs systèmes d'exploitation (Windows 2000 et XP, Linux avec FreeSwan [17], etc.) implémentent nativement le protocole IPSec. Tous nos essais ont été effectués avec ces implémentations de IPSec qui ont l'immense avantage de ne nécessiter aucune installation de produit supplémentaire.
Nous avons utilisé IPSec pour construire un VPN. L'ordinateur de l'utilisateur nomade tournant sous Windows 2000/XP ou Linux devient un élément d'un VPN. Même en étant connecté à l'extérieur il est considéré comme faisant partie du réseau interne. Il n'y a alors plus de différence entre ce que l'in peut faire en étant connecté sur le réseau interne ou à l'extérieur.
L'authentification s'effectue à l'aide d'un certificat client. Comme IPSec se situe au niveau de la couche réseau et est géré directement par le noyau du système d'exploitation le certificat est associé à l'ordinateur et non à l'utilisateur. Ce qui est authentifié est donc l'ordinateur. Un mot de passe n'est pas demandé à l'utilisateur pour déverrouiller la clé privée ce qui signifie qu'il faut faire confiance à la machine et au système d'exploitation. Avec Windows 2000/XP il n'est pas possible d'utiliser un dispositif physique comme une carte à puce pour ranger le certificat et la clé privée associée. On peut donc être amené à ajouter par ailleurs une authentification de l'utilisateur.
Afin d'établir une connexion IPSec le trafic pour les protocoles IP ESP (50), AH(51) et UDP port 500 doit être autorisé sur les différents routeurs et coupe-feu tout au long de la route. En outre les paquets IP fragmentés doivent être acceptés. La traduction d'adresse (« Network Addresses Translation ou NAT) posent de sérieuses difficultés.
Etant donné les nombreuses contraintes pour implémenter IPSec directement au niveau du poste de travail de l'utilisateur on peut se demander si cette technologie n'est pas plutôt réservée aux opérateurs de réseau souhaitant offrir des VPN à leurs clients et à quelques utilisateurs très avertis.

9 802.1x

Le nouveau standard Ethernet 802.1x est une réponse au besoin d'authentifier les machines ou les utilisateurs connectés sur un réseau local. Aujourd'hui les trop nombreuses faiblesses dans la sécurité du Wi-Fi [18] en limitent son développement. Les différents acteurs du marché poussent vers l'adoption de ce nouveau standard afin conserver leur business. Ce standard peut aussi être utilisé pour sécuriser les connexions Ethernet câblées.
802.1x [19] définit 3 rôles : requérant[5] (« supplicant »), certificateur (« authenticator ») et serveur d'authentification (« authentication server »). Le requerrant est le poste travail demandant un accès au réseau. Le certificateur est l'unité un commutateur ou borne d'accès sans fil fournissant la connexion au réseau. Un port sur l'unité peut avoir 2 états : non autorisé ou autorisé. Tant que le client n'est pas authentifié le port est dans l'état non autorisé et le seul le trafic permis entre le requerrant et le certificateur est celui nécessité par l'authentification. Le certificateur transmet la requête d'authentification au serveur d'authentification (RADIUS) en utilisant le protocole EAP [20]. Si le serveur d'authentification valide la demande, le port est commuté dans l'état autorisé et alors la station de travail est autorisée à avoir un accès complet au réseau.
Plusieurs méthodes d'authentification sont disponibles. Parmi celles-ci EAP-TLS et EAP-TTLS utilisent des certificats.

[5] [5] Les termes requérant et certificateur font partie du vocabulaire juridique. Mais ils ont l'inconvénient de rester trop loin des termes d'origine (suppliant est vraiment inapproprié et « authentificateur » n'existe pas).

Le protocole 802.1x apporte aussi un mécanisme d'échange pour les clés servant à sécuriser les transmissions. C'est un immense progrès par rapport au WEP défini dans le protocole 802.11.
Le serveur RADIUS peut aussi fournir un certain nombre d'informations en fonction de qui est authentifié. Parmi ces informations on peut avoir le numéro de VLAN et avec certains modèles de commutateurs des règles de contrôle d'accès (ACL) pour le port concerné [21].
A notre connaissance les systèmes implémentant de façon native le standard 802.1x sont actuellement : Windows 2000 SP4, Windows XP ainsi que MacOS X, FreeBSD, OpenBSD, Linux avec open1x [22].
A la suite du déménagement du laboratoire dans de nouveaux locaux et avec la disponibilité de commutateurs de génération récente, nous avons commencé à utiliser le 802.1x pour gérer dynamiquement les VLAN et établir des règles d'accès au réseau. Nous considérons que l'utilisation du 802.1x est absolument indispensable pour les accès Wi-Fi et que c'est un excellent moyen pour contrôler les accès Ethernet filaire des ordinateurs portables au réseau local dans notre laboratoire.
Jusqu'à maintenant nous avions basé notre modèle de sécurité sur le couple fixe (port sur le commutateur, adresse Ethernet) auquel on attribue des éléments comme l'adresse IP et le numéro de VLAN. Ce n'est pas complètement sûr car l'adresse Ethernet peut aisément être usurpée. La gestion est relativement complexe, chaque fois qu'une nouvelle machine est connectée, une machine existante est déplacée d'une pièce à une autre il faut mettre à jour les différentes tables, générer et appliquer les nouvelles configurations des commutateurs. 802.1x permet une authentification forte de l'ordinateur ou de l'utilisateur ce qui est largement plus sûr que l'adresse Ethernet et d'affecter des droits d'accès au réseau en conséquence.

10 Comparaisons

Dans le tableau suivant nous avons essayé d'évaluer avec les notes suivantes (mauvais, * passable, ** bien, *** excellent) les outils vus précédemment en fonction des critères suivants :

- Le niveau global de sécurité.
- La simplicité d'usage et les fonctionnalités.
- La facilité d'installation.
- La disponibilité sur les différentes plates-formes.
- La polyvalence ou la possibilité d'offrir un large spectre d'application.

Tous ces aspects sont abordés du point de vue de l'utilisateur. Nous considérons en effet qu'il doit être placé au centre du système d'information.

	Sécurité	Usage	Installation	Disponibilité	Polyvalence
Webmail	**	**	***	***	*
Courriel sécurisé	**/***[6]	***	**	**	*
Intranet	***		**	***	**
Portail captif	*	*	**	***	**
IPSec	***	**	*	*	***
802.1x	*		*	*	***

Tableau 1: Comparaison entre différents outils

11 Choix d'une méthode

	Webmail WebFTP	TLS	IPSec	SSH
Courrier (libre service)	***			*
Courrier (machine personnelle)	**	***	*	*
Transfert de fichiers	*		**	***
Intranet		***	*	
Tout protocole			***	*

Table 2: Critères pour choisir un outil

[6] ** sans certificat client, *** avec

12 Quelques réflexions sur l'usage des certificats pour l'authentification

Le modèle classique avec identifiant et mot de passe permet au serveur d'authentifier le client mais la réciproque n'est pas vraie. Le client n'a aucun moyen d'authentifier le serveur, ce qui ouvre la porte à toute une classe d'attaque où le pirate cherche à se faire passer pour le serveur. Ce modèle est notoirement insuffisant, aujourd'hui une authentification mutuelle est indispensable.
Quasiment toutes les méthodes pour authentifier le serveur reposent sur le chiffrement à clés asymétriques. On voit mal comment on pourrait utiliser un mot de passe, sachant qu'un serveur peut avoir de nombreux clients. Soit le client a récupéré par un canal sûr la clé publique du serveur comme avec SSH, soit il utilise le certificat du serveur certifié par une autorité de confiance. En terme de déploiement de certificats cela reste assez facile à gérer. Le nombre de serveurs est relativement limité, ils sont administrés par des personnes compétentes. Côté client, la seule opération à effectuer est l'installation, le cas échéant, du certificat de l'autorité racine.
La combinaison certificat pour le serveur, mot de passe pour le client est probablement celle qui est aujourd'hui la plus répandue. Elle offre un niveau de sécurité satisfaisant dans bon nombre de cas.
L'utilisation d'un certificat pour authentifier le client est relativement contraignante car elle nécessite une infrastructure de gestion de clé. Souvent seule la sécurité supplémentaire apportée par une carte à puce ou un dispositif équivalent justifie l'utilisation d'une authentification du client par certificat. Dans un certain nombre de cas où le certificat et sa clé privée sont rangés sur disque, il n'est même jamais exigé de mot de passe pour déverrouiller la clé privée, la sécurité est alors inférieure à celle d'une authentification par mot de passe.
L'utilisation d'un certificat pour authentifier le client permet dans la mesure où toutes les applications utilisent le même certificat de réaliser simplement une authentification unique (« Single Sign On » ou SSO).

13 Conclusions

Si le développement et le déploiement d'une IGC est un grand projet qui exige beaucoup de ressources, un fois acquis l'utilisation des certificats s'avère une excellent méthode pour sécuriser les accès des utilisateurs nomades au réseau.
Différentes solutions sont disponibles. Chacune a ses avantages et inconvénients. Il n'y a pas de panacée, uniquement des réponses partielles qu'il faut combiner en fonction des besoins.

Références

[1] W.Diffie and M.E.Hellman, "New directions in cryptography," IEEE Trans. Inform. Theory, IT-22, 6, 1976, pp.644-654.
[2] Rivest, R.L., Shamir, a. and Adelman, L. "*On Digital Signatures and Public Key Cryptosystems*," MIT Laboratory for Computer Science Technical Memorandum 82, April 1977
[3] IGCCNRS http://www.urec.cnrs.fr/igc/Doc/IGC_docs.html
[4] SquirrelMail http://www.squirrelmail.org/
[5] JAWMAIL http://www.jawmail.org/
[6] Horde Imp http://www.horde.org/imp
[7] Albert Shih WebFTP http://www.institut.math.jussieu.fr/
[8] Horde Gollem http://www.horde.org/gollem
[9] Netscape http://wp.netscape.com/eng/ssl3/
[10] TLS http://www.ietf.org/html.charters/tls-charter.html
[11] Cyrus http://asg.web.cmu.edu/cyrus/
[12] SSH http://www.ssh.com/
[13] openssh http://www.openssh.com/
[14] Captive Portal http://www.personaltelco.net/index.cgi/CaptivePortal
[15] gwauth http://www.lmcp.jussieu.fr/~morris/gwauth
[16] IP Security Protocol (ipsec) http://www.ietf.org/html.charters/ipsec-charter.html
[17] FreeSwan http://www.freeswan.org/
[18] Nancy Cam-Winget, Russ Housley, David Wagner, Jesse Walker. Security Flaws in 802.11 Data Link Protocols. Communications of the ACM May 2003/Vol. 46, N° 5
[19] 802.1x - Port Based Network Access Control http://www.ieee802.org/1/pages/802.1x.html
[20] Extensible Authentication Protocol (eap) http://www.ietf.org/html.charters/eap-charter.html
[21] Benjamin Dexheimer, Roland Dirlewanger, François Morris. Les réseaux privés virtuels. Tutoriel JRES 2003, Lille.
[22] Open Source Implementation of IEEE 802.1x http://www.open1x.org/