Hacking –éthique–

Bruno Martin

Université Côte d'Azur

M1 Informatique

Plan

Termes

- **Penetration test :** méthode d'évaluation de la sécurité d'un hôte ou réseau en simulant une attaque. Pour ça :
  - rechercher les points d'accès
  - rechercher les vulnérabilités

  selon différentes approches :
  - **Black box (covert)** : pas de connaissance de l'infrastructure ; effacer ses traces
  - **White box (overt)** : infrastructure connue, avec RSSI
  - et les variantes entre les deux (grey box).
- **Forensic** : Terme adapté de l'anglais ≪computer forensics≫, l'expression ≪ investigation numérique ≫ représente l'utilisation de techniques spécialisées dans la collecte, l'identification, la description, la sécurisation, l'extraction, l'authentification, l'analyse, l'interprétation et l'explication de l'information numérique.

Phases du PenTest

- Pre-engagement interaction : négociation avec client : "contrat"
- Intelligence gathering : récupération de toutes les infos possibles sur le client (réseaux sociaux, scan, footprint,...)
- Threat modeling : utiliser les infos de l'IG pour identifier les vulnérabilités, choisir les attaques en fonction des buts cherchés
- Vulnerability analysis : trouver les attaques possibles en fonction de l'analyse des ports et des vulnérabilités,...
- Exploitation : réalisation d'exploits
- Post exploitation : attaques en whitehat
- Reporting : rapporter le détail des opérations menées

Bug bounty

Un bug bounty est un programme proposé par de nombreux sites web (depuis 1995 avec Netscape) et développeurs de logiciel qui permet à des personnes de recevoir reconnaissance et récompense après avoir reporté des bugs, surtout ceux concernant des exploits et des vulnérabilités.
Il y a bien sûr des règles à respecter et chaque Bug Bounty doit énoncer clairement les limites que le hacker ou l'expert ne doit pas franchir, mais en général, comme ça se passe sur des services en production, il vaut mieux éviter de tout casser si on veut sa récompense ;-).
En 2015, M. Litchfield dit avoir gagné plus de 300000$ en trouvant des failles.
Source korben

Avertissement

- Utilisation des outils décrits plus loin est ILLEGALE !
- Interdiction FORMELLE d'utiliser ces outils ailleurs que dans les salles prévues pour cet usage
- Ils peuvent servir à sécuriser un réseau et à le pirater
- Ne JAMAIS les utiliser ailleurs que sur un LAN privé
- Pas dans le cadre de la fac : Charte informatique
- Outils d'audit=outils d'attaques=armes
  - Pas les pointer sur des cibles réelles
  - Prendre toutes les précautions
  - Demander l'autorisation de l'admin et FAI
- Perpétrer des actes de piratage est répréhensible
  - Peut vous coûter votre carrière
  - C'est TRES sérieux

How hackers do it

- Inspiré de « How hackers do it :..Tricks, Tools, and Techniques » A. Noordergraaf, Sun Blueprints..May 2002
- Actualisé pour la partie « outils »
- Grandes étapes :
  - Identifier la cible
  - Collecter des infos
  - Lancer l'attaque
  - Couvrir ses traces
  - Maintenir l'accès

Intelligence gathering

- un bon hacker (bidouilleur) programme un outil pour scanner (explorer) le réseau
- il le publie sur Internet
- un script kiddie (novice) l'utilise pour trouver des systèmes vulnérables ou des points d'accès

Identifier la cible

- Connaître c'est faire la moitié du chemin
  - Si les défenses sont connues, plus facile de planifier l'attaque
- Moyens
  - Détection de l'hôte (ping)
  - Recherche des services (port scan, banner grabbing)
  - Détection de la topologie
  - Traceroute, wardialing, wardriving
  - Détection d'OS par son fingerprint
  - Sources publiques (whois, dns, web, annuaires)
  - Social engineering, OSINT

Outils de base

- `Nslookup/dig` : Résolution de noms de domaine
- Ping :
  - Vérifier quelles sont les machines en ligne
  - Trouver les adresses de broadcast
- `Traceroute` :
  - Combien de routeurs jusqu'à la cible ?
  - à la main par envoi de paquets tcp en changeant le TTL
- `Finger` :
  - collecte les informations sur les utilisateurs
  - `finger alice@host.target` : infos sur alice
  - `finger @host.target` infos utilisateurs connectés
- `netcat` ou `nc`
  - Outil multifonctions (aussi appelé TCP/IP swiss armyknife) utilisable à différentes fins
  - récupérer les bannières des serveurs (banner grabbing)
  - `nc -v -n host.target 22` renvoie la version de `ssh` qui tourne sur la machine (SSH-1.99-OpenSSH_5.1)

Scanner de ports : `nmap`

Un port scanner peut parcourir une grande plage d'adresses IP et retourner les ports ouverts (donc les services accessibles) ainsi que les version d'OS

Description

Nmap (”Network Mapper”) is a free open source utility for network exploration or security auditing. It was designed to rapidly scan large networks, although it works fine against single hosts. Nmap uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics. Nmap runs on most types of computers and bothconsole and graphical versions are available. Nmap is free software, available with full source code under the terms of the GNU GPL.

`nmap`, modes de fonctionnement

vanilla tentative de connexion sur tous les ports

strobe cible certains ports spécifiques

fragment packets limitation à des paquets fragmentés (pour traverser certains fw)

udp recherche des ports `udp`

sweep connexion sur le même port d’un ou plusieurs PC

FTP bounce imite le fonctionnement d’un serveur `ftp` pour paraître légitime

stealth permet d’augmenter le discretion en empêchant partiellement le fonctionnement des mécanismes de log

Scanner de ports : `nmap`

OS detection

Tehniques standard : par banner grabbing ; sinon

- connexion `smtp`, `snmp` ou `telnet` pour examiner les réponses du serveur
- dans `nmap` fingerprint de la pile `tcp/ip` qui permet d’identifier la réponse du système à des paquets `tcp` avec des drapeaux particuliers

Collecte d'informations

- Travailler de façon systématique en notant tout
- **Objectif :** Avoir une meilleure connaissance du réseau attaqué que son admin (en tout cas, plus à jour)
- Pour chaque IP, déterminer :
  - Son nom de domaine,
  - Si la machine est en vie (ping, arp)
  - Quels ports sont ouverts
  - Pour chaque port
    - Quels services offerts
    - Quel serveur, version, service pack
  - Quel OS
  - La machine sert-elle de routeur ?
  - Service ouvert plus indicatif ?

Vulnérabilités

- le novice utilise ensuite une liste d'IP vulnérables pour accéder au système
- selon les faiblesses, il peut éventuellement créer/utiliser un compte ou un accès privilégié
- il l'utilise pour acquérir de nouveaux privilèges et pour pirater de nouveaux systèmes connectés à sa 1re victime
- exemple : se faire passer pour une machine du réseau attaqué (avec `wireshark` ou `ettercap`)

Vulnérabilités

- Une fois à l'intérieur :
- trouver
  - les utilisateurs, les mots de passe à casser
  - Les fichiers intéressants
  - Les connexions réseau actives
  - Les tables de cache arp
  - Les indications sur d'autres machines

Outils d'audit : sniffers

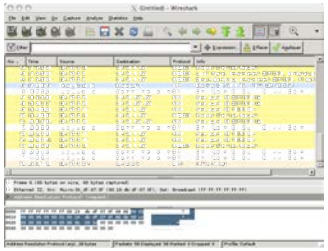
Packet sniffers : logiciels d'écoute des données non-chiffrées d'un LAN. Servent à

- intercepter mdp (ou autre info) qui transite en clair
- résoudre des problèmes réseaux en visualisant le traffic
- la rétro-ingénierie réseau

`http://fr.wikipedia.org/wiki/Packet_sniffer`

Attention à l'utilisation selon l'architecture du LAN (hub ou switch) !

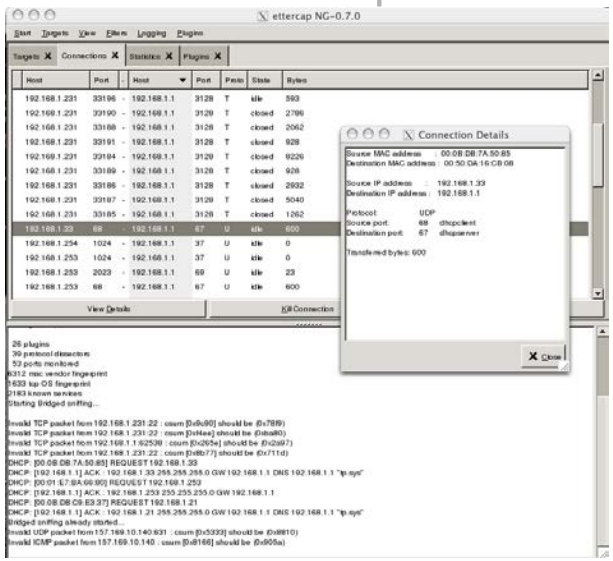
Obtenir un accès : Wireshark



C'est un sniffer : programme qui lit tout le trafic sur le LAN en passant l'interface réseau en mode de promiscuité
Ainsi, la carte réseau transmet au sniffer tout le trafic réseau
Autres sniffers : snoop (WIN) qui permet de récupérer des mots de passe transmis pour une connexion telnet, ftp, imap,pop

Cliquer ici pour la video (telnet)

Ettercap



Ettercap- description

- suite for MIM attacks on LAN
- features sniffing of live connections, content filtering on the fly and many other interesting tricks
- supports active and passive dissection of many protocols (even ciphered ones) and includes many features for network and host analysis

Cliquer ici pour la video (imap)

Rappels sur ethernet

L'acheminement des trames ethernet s'appuie sur la notion de compétition pour l'accès au media.

Chaque hôte écoute et attend un ≪ silence media ≫ avant d'émettre. Si un signal transite sur le media, les émetteurs attendent que le media soit libre.
Quand le canal est libre, un hôte émet.
Quand 2 hôtes émettent simultanément, il y a **collision**. Dans ce cas, les 2 hôtes détectent la collision et envoient un signal de collision qui empêche l'émission pour une durée aléatoire.

Un **domaine de collision** est une région du réseau au sein de laquelle les hôtes partagent l'accès au media.

Fonctionnement hub (concentrateur)

Quand un paquet est reçu, il est propagé sur toutes les interfaces sauf celle de l'émetteur. Hub ne délimite ni les domaines de collision ni les domaines de broadcast.

Permet à la carte réseau d'un hôte d'accepter tous les paquets qu'elle reçoit, même s'ils ne lui sont pas destinés (mode de promiscuité).

Détection du mode de promiscuité :

- augmentation charge de l'hôte qui traite tous les paquets et augmente la latence du réseau
- détection avec `detectpromisc`

Inconvénients hub

Réseau avec beaucoup d'hôtes, problèmes de performance :

- **disponibilité :** partage de la bande passante ; un hôte peut monopoliser tout le trafic (gros transfert)
- **latence :** (temps nécessaire à un paquet pour atteindre sa destination). Avec des hubs on attend une opportunité de transmission pour éviter les collisions. Latence croît en fonction du nombre d'hôtes du réseau.
- **défaillance :** plus sensible aux pannes ou aux mauvaises configurations de la vitesse de transmission

Un switch (commutateur) résout ces problèmes.

Avantages switch (commutateur)

- Switch améliore la disponibilité et la latence en délimitant les domaines de collision
- Chaque hôte connecté dispose de toute la bande passante.
- un paquet qui arrive sur un port du switch n'est retransmis que sur le port auquel le destinataire est connecté
- Signaux de collision non retransmis par les switches

Fonctionnement switch

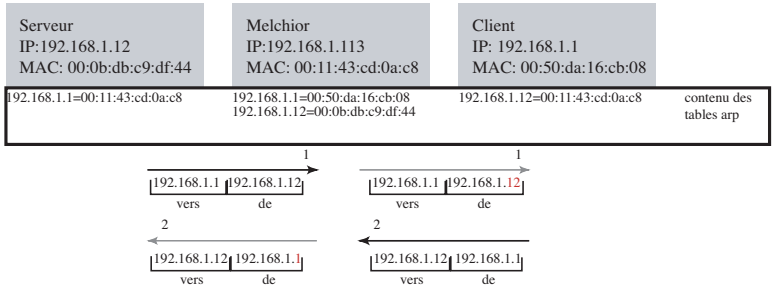
Un paquet qui arrive dans le switch est mis dans le buffer. L'adresse MAC du paquet est lue et comparée à la liste des MAC connues rangées dans la **table de lookup**.
3 modes d'acheminement :

- **cut through :** lecture des 6 octets MAC dest. et routage direct sans traitement vers le port du destinataire
- **store and forward :** mise en mémoire et traitement du paquet avant son acheminement (rejeter les paquets mal formés, gérer les messages de collision)
- **fragment free :** analogue au cut through mais lit les 64 premiers octets avant le routage. (cela limite les erreurs de collision qui arrivent souvent sur les 64 premiers octets)

Le mode le plus utlisé est le store and forward.

Ecoute sur réseau switché

Mode promiscuité inutile. Utiliser arp spoofing/poisoning :



2 outils : `dsniff` et `ettercap` disponibles sur Kali. Nécessaires pour utiliser des outils d'écoute du réseau (`dsniff`, `wireshark`, `tcpdump`).

Utilisation `dsniff`

Contient un utilitaire `arpspoof`.
Pour mener à bien l'attaque :

- activer l'**IP forwarding** sur M ; l'IP forwarding permet de faire transiter des paquets d'une interface réseau à une autre. La machine va servir de ≪ routeur ≫
- activer le spoofing dans les 2 sens :
  ```
  arpspoof -t IP1 IP2 & >/dev/null
  arpspoof -t IP2 IP1 & >/dev/null
  ```
- terminer par :
  ```
  killall arpspoof
  ```


Utilisation `ettercap`

`ettercap` a 2 modes de fonctionnement : interactif (interface ncurses ou Gtk) ou en CLI.
Pour mener à bien l'attaque :

- l'IP forwarding est automatiquement activé par `ettercap`
- empoisonner tout le trafic par :
  `ettercap -T -q -M ARP // //`
  - `-T` : choix type interface
  - `-q` : mode silencieux (quiet)
  - `-M ARP` : attaque MIM type `arp`
  - `// //` : de la source vers la destination
- empoisonner une cible (`IP1`) par :
  `ettercap -T -q -M ARP /IP1/ //`
  redirige tout le trafic entre `IP1` et le reste du réseau

Outil d'audit : crack, hashcat

- Erreurs d'Unix :
  - `/etc/passwd` lisible par tous, même si non inversible
  - Même algorithme de chiffrement sur toutes les machines
- Par combinaison (sans dictionnaire), la puissance actuelle des machines permet de découvrir des mots de passe jusqu'à plusieurs caractères
- Crack à partir de mots de dictionnaires :
  - Ajoute des mots provenant de `/etc/passwd` (nom, ...)
  - Crée de nouveaux mots (`cle+`, `Cle`, `elc`, ...)
  - Chiffre le mot, le compare avec le contenu de `/etc/passwd`
  - Mémorise les mots de passe testés
- Configurable :
  - Nouvelles règles pour générer de nouveaux mots
  - Ajout de dictionnaires (ciblés linguistiquement)
  - Peut travailler sur plusieurs fichiers passwd
  - Peut envoyer un message aux utilisateurs

Robustesse du mot de passe

Howto

- Un peu de psychologie
  - Un utilisateur se connecte
    - sur un serveur imap (pas imaps)
    - Sur un serveur pop
    - par telnet ou ftp
- Quelques outils d'observation passive
- Et le tour est joué :
  - Fred dispose maintenant peut-être d'un couple login/ password valide. . .
  - D'ailleurs, ettercap permet aussi théoriquement de récupérer des mots de passe ssh1 entre autres choses
  - D'autres outils plus spécifiques font la même chose !

Encore plus fort

- Tout est raisonnablement sécurisé sauf l'OS de certaines machines.
- Que fait le pirate ?
- Il s'attaque à la machine la plus faible

Règle :

La sécurité du système est celle de son maillon le plus faible.

Comment trouver cette faiblesse ?

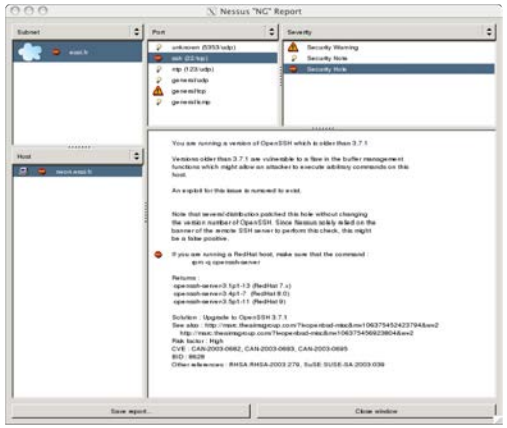
- Au moyen d'un scanner de vulnérabilité
- Par analyse des ports d'entrée, le scanner de vulnérabilité trouve l'hôte ou le service le plus faible
- C'est ce que fait nessus ou OpenVAS !

Scanner de vulnérabilités – Nessus

Le projet ≪ Nessus ≫ a pour but de fournir à la communauté internet un un scanner de vulnérabilité à distance gratuit, puissant, actualisé et facile à utiliser, logiciel permettant d'auditer à distance un réseau donné et de déterminer si quelqu'un peut s'y introduire ou l'utiliser à mauvais escient. Nessus ne prend rien pour acquis. C'est-à-dire qu'il ne considère pas qu'un service donné fonctionne sur un port fixe ; un serveur web sur le port 1234 sera détecté et sa sécurité testée.
Nessus est rapide, fiable et possède une architecture modulaire qui vous permet de l'adapter à vos besoins. Nessus fonctionne sur les systèmes de type Unix (MacOS X, FreeBSD, Linux, Solaris et autres) et une version Windows est disponible.

Nessus : cibler

Nessus : scanner



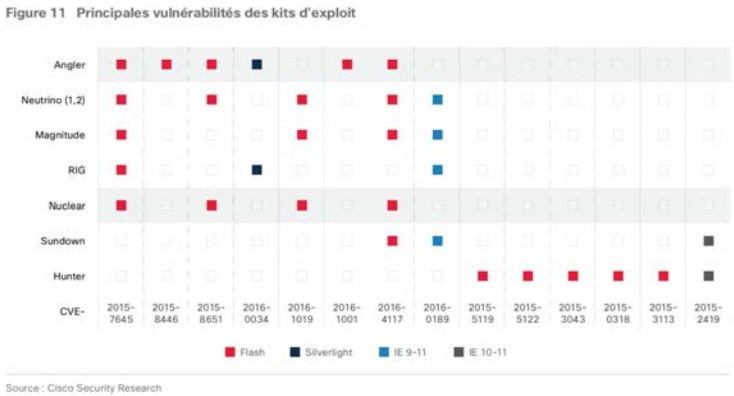
Nessus - OpenVAS

Nessus est devenu payant (cher) en 2005.
OpenVAS représente la branche « libre » de nessus. Contenait en 2011 plus de 23 000 tests de vulnérabilité, reliés à la base “Common Vulnerabilities and Exposures” CVE qu’on peut interroger
Il est possible d’ajouter des plugins dans le langage NASL, comme dans nessus.
`http://www.openvas.org/`

Attaquer

- généralement au moyen de `rootkits`
- un `rootkit` est un terme qui décrit un ensemble de scripts et d’exécutables qui permettent à un pirate de cacher ses agissements et d’obtenir un accès privilégié au système :
  - modifie les `logs`
  - modifie les outils système pour rendre la détection du piratage difficile
  - crée une trappe d’accès cachée
  - utilise le système comme point d’entrée sur les autres hôtes du LAN

Kits d'exploits



Voir rapport CISCO

Framework Metasploit

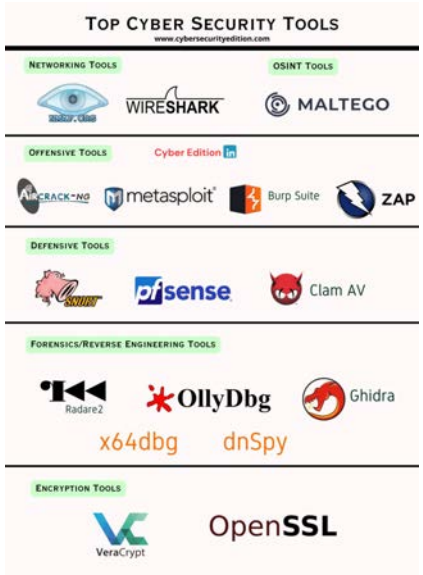
Metasploit (écrit en `ruby`) est un framework qui permet :

- de collecter le résultat des différents scanners (port, vulnérabilité,...)
- d'automatiser (et de rejouer) des attaques contre des vulnérabilités identifiées (et d'en ajouter)

outil très puissant mais compliqué à utiliser (gui : armitage)
`https://www.youtube.com/watch?v=AG_MeOsnQwM`

Et pour le wifi ?

- On procède de même :
  - Cibler avec kismet (802.11) qui permet de détecter, de sniffer
  - attaquer avec aircrack qui permet de casser les mots de passe WEP et WPA-PSK une fois qu'on a récupéré assez de traffic.



Exemple pratique

Coût

- 800 k€ : coût moyen d'une violation de sécurité
  - 330 k€ pour une entreprise de taille intermédiaire
  - 1,3 M€ pour une grande entreprise
- 9 semaines pour réparer les dégâts
- Préconisation : 5% du budget à la cybersécurité

[source IBM]

Modèle R = V.M.C

- **Vulnérabilité** : faiblesses connues de l'architecture de sécurité (trop de points d'accès, faible authentification,…)
- **Menace** : ce contre quoi on cherche à se défendre (DoS, …)
- **Coût** : impact financier
- **Risque** : quantification des menaces potentielles et des vulnérabilités

Test de vulnérabilité

Détermine la surface d'attaque

Greenbone Security Assistant

Menaces

Modèle de menace

- Vulnérabilité menacée
- Par quel moyen ?

Top 5 -2019-

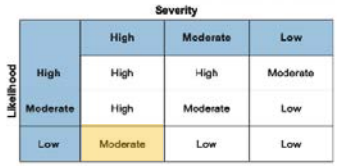
- DNS Hijacking (→ MiTM)
- Rançongiciels
- Remote Access Trojan
- Office 365 Phishing
- Digital Extorsion Scams

Evaluation des risques

1. Chiffrer la valeur d'une ressource
2. inventorier les avantages (logiciel, matériel, biens, RH,…)
3. identifier les menaces
4. identifier les vulnérabilités
5. analyser et mettre à jour les mécanismes de contrôle
6. évaluer l'impact et la probabilité de scenarii d'attaques
7. prioriser les risques selon le coût de leur prévention vs la valeur
8. documenter, informer et former

Matrice des risques

- **Elevé** : apporter des correctifs au plus vite
- **Modéré** : appliquer des mesures dans un délai raisonnable
- **Faible** : accepter le risque ou le réduire



Rapport Terranova phishing :
14% des destinataires ont divulgué leurs identifiants
probabilité : faible
gravité : élevée
Décision : risque modéré

Sensibiliser
Former
Mener une campagne test

Quel procédé ?

1. reconnaissance (récupérer infos sur la cible)
2. planification de l'attaque (le maillon faible est trouvé)
3. hameçonnage (campagne de phishing)
4. reconnaissance interne (l'attaquant est à l'intérieur)
5. attaque véritable (sur les cibles les plus intéressantes)
6. masquer ses traces

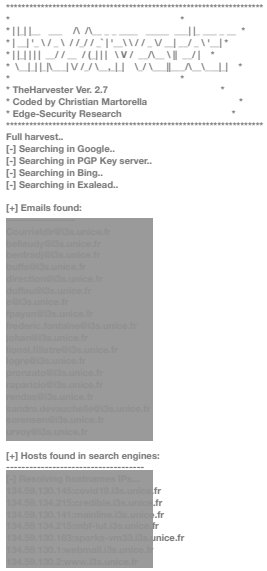
Quels outils ?

+ tous ceux qu'on peut …
programmer,
… acheter,
… imaginer

Reconnaissance

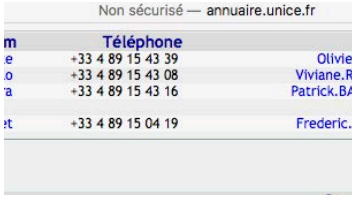
- données publiques :
  `theHarvester`

- J'ai quelques mails, j'en cherche d'autres
- J'ai des noms de machines : celle du mail !



D’autres mails

- simplement par l’annuaire !



- je mène une campagne de phishing : `Social Engineering Toolkit(SET)`
- `clonage du site qu’on va héberger`

- un des 14% d’utilisateurs se connectera au clone et sera redirigé vers le vrai site
- j’ai son mot de passe

A l’aide Google

- Je trouve aussi les info sur le VPN grâce à Google (parmi les identifiants récupérés, il y en a qui vont marcher)
- Là, je suis vraiment potentiellement à l’intérieur avec un compte utilisateur

Aller plus loin

- Un de mes utilisateurs utilise le même mdp pour le mail et pour le SSO
- J’accède à ses mails et aux applis web auxquelles il a accès (et peut-être à une machine accessible)

Plus technique

- Défaut de PKI (attaque MiTM possible)


```
maMachine:~ moi$ openssl s_client -connect imap.unice.fr:993 -status -verifyCApath /etc/ssl
---
Certificate chain
 0 s:C = FR, postalCode = 06100, ST = Provence-Alpes-Côte-d'Azur, L = Nice, street = Grand Château, street = 28 avenue Valrose, O = Université Côte d'Azur, CN = pop.unice.fr
---
Server certificate
-----BEGIN CERTIFICATE-----

-----END CERTIFICATE-----
subject=C = FR, postalCode = 06100, ST = Provence-Alpes-Côte-d'Azur, L = Nice, street = Grand Château, street = 28 avenue Valrose, O = Université Côte d'Azur, CN = pop.unice.fr

issuer=C = NL, O = GEANT Vereniging, CN = GEANT OV RSA CA 4
---
SSL handshake has read 7369 bytes and written 450 bytes
Verification error: self signed certificate in certificate chain
```

Autre technique

- Injection SQL (sur différents sites trouvés par `OWASP` permet éventuellement de récupérer des mdp hachés)
- location serveur 1$/h avec Nvidia Tesla K80 qui peut trouver la préimage de l'empreinte. Testé (dans un article) 2h : 48% des mdp cassés (14 millions de mdp, donc 6,7 million retrouvés)

Et quand tout a échoué

- Il faut avoir fait des sauvegardes. . ..
- Pouvoir tout réinstaller et sans perte
- C'est le plan de reprise d'activité

Quelques remarques

- Une attaque se porte sur la plus grande vulnérabilité
- 1/3 des attaques est interne
- On ne peut rien faire contre une vulnérabilité 0-day
- On passe d'un stade « artisanal » à un
- La technicité augmente

Les deux piliers de la cybersécurité

- **Prévenir** : éviter de se faire attaquer (au moyen de firewall, programmes antivirus, penser à faire les mises à jour, vérifier les mails entrants
- **Reprise d'activité** : Il n'est pas possible de prévenir tous les risques. Des attaques fructueuses peuvent toujours être menées. Il faut prévoir des sauvegardes et un plan de reprise d'activité

Les 10 règles de la sécurité

- Sécuriser les points faibles
- Opérer en profondeur
- Bien gérer les cas d'erreur
- Principe du strict minimum
- Cloisonner
- Rester simple
- Encourager le secret
- Il est difficile de garder un secret
- Rester méfiant
- Utiliser les ressources de la communauté