SÉCURITÉ DES SERVEURS ET DES APPLICATIONS

Méthodologie d'attaques

K. Poutrain

02/04/2015

7ème journée du réseau ARAMIS – Lyon

Plan

Les outils : Kali Linux & Owasp BWA

Méthodologie d'attaque : les étapes

Étude du cas : Les injections SQL

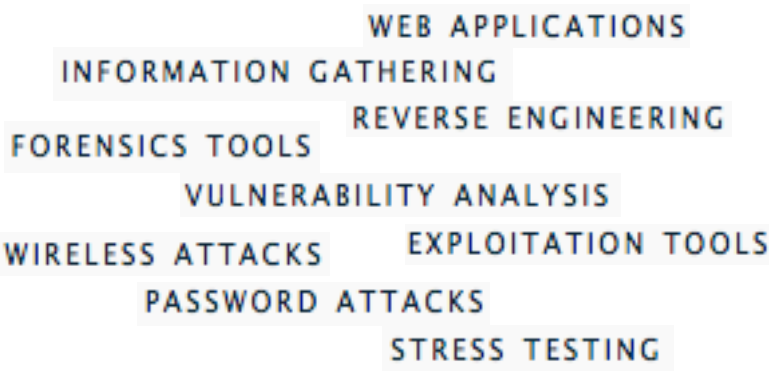
Démos

Les outils du test-lab

KALI LINUX™

"the quieter you become, the more you are able to hear"

Plateforme d'attaque
Distribution Linux (outils)



https://www.kali.org/

KALI LINUX ™ is a trademark of Offensive Security.

VS

OWASP BWA
(Broken Web Applications)

Plateforme web vulnérable
Machine virtuelle
Applications « école »
Applications réalistes vulnérables
Applications réelles vulnérables
(ancien wordpress, Joomla,…)

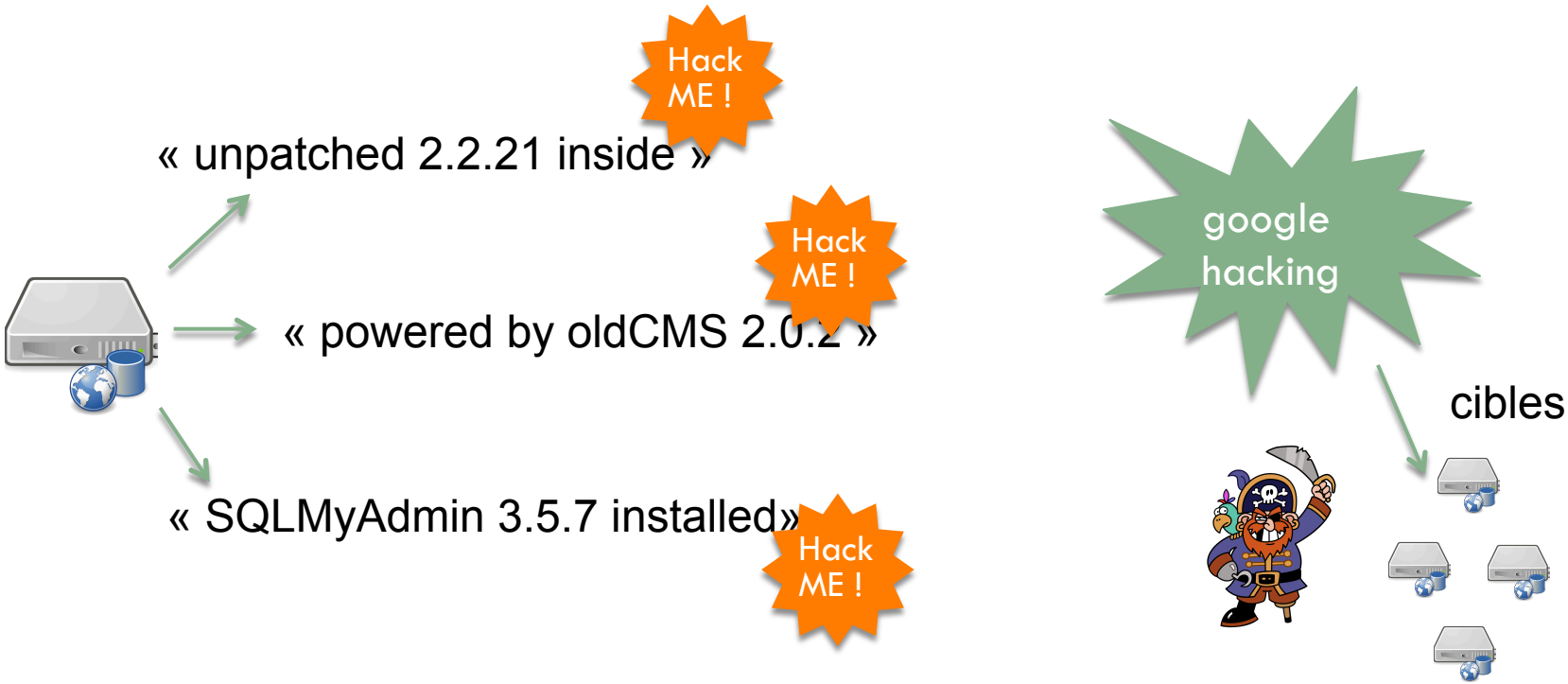
https://www.owasp.org/index.php/OWASP_Broken_Web_Applications_Project

METHODOLOGIE D'ATTAQUE

Quelle(s) méthode(s) d'attaque ?

METHODOLOGIE D'ATTAQUE

CAS 1 : attaque opportuniste



LIMITER LA DIFFUSION D'INFORMATIONS !

METHODOLOGIE D'ATTAQUE

CAS 2 : attaque ciblée

- objectifs divers :
  - vol de données
  - altération d'information (défiguration)
  - mise en ligne de sites frauduleux (vente, phishing)
  - envoi de spams, obtention d'un shell, rebond…

=> Technique méthodique, exhaustive

METHODOLOGIE D'ATTAQUE : LES ÉTAPES CLES

- Reconnaissance (collecte d'informations)
- Cartographie de la surface d'attaque
- Exploitation (attaque)
- Elévation de privilège
- Conservation d'accès

RECONNAISSANCE

La plus importante des étapes !

- identifier le contenu de l'application ;
- analyser son comportement ;
- en comprendre le fonctionnement ;
- étudier son contexte (qui l'utilise, qui la développe ?).

=> cartographier l'application

- représentation structurelle (arborescence des fichiers, liens) ;
- représentation fonctionnelle (logique de l'application).

RECONNAISSANCE : CARTOGRAPHIE

Robot d'indexation automatique (spider/ crawler) ?

Limites :

- liens dynamiques JavaScript ;
- liens accessibles après remplissage d'un formulaire…) ;
- risque de fermer la session (lien « logout ») ;
- attention aux actions dangereuses ! (reboot, réinitialiser la BDD…).

CARTOGRAPHIE PILOTÉE

Cartographie pilotée par l'utilisateur.

Proxy d'interception :

BurpSuite

OWASP-ZAP (Zed Attack Proxy)

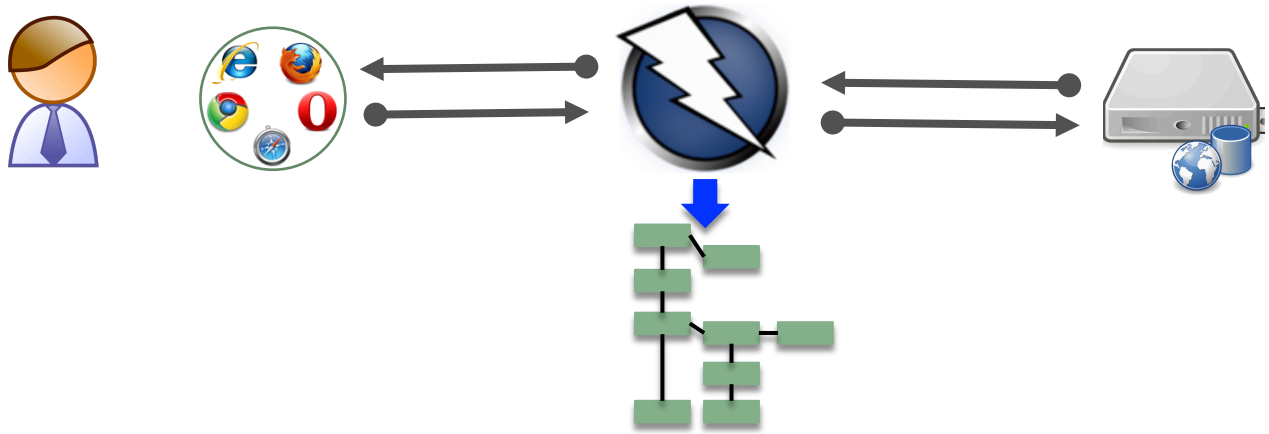
CARTOGRAPHIE PILOTÉE

1 - L'utilisateur réalise un **parcours exhaustif** des liens dans son navigateur.

Le proxy :

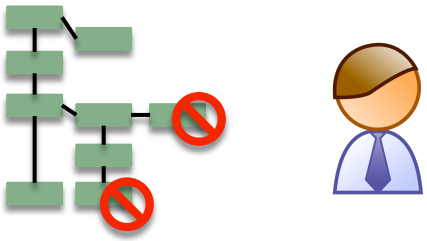
analyse les requêtes/réponses

construit la carte de l'application.

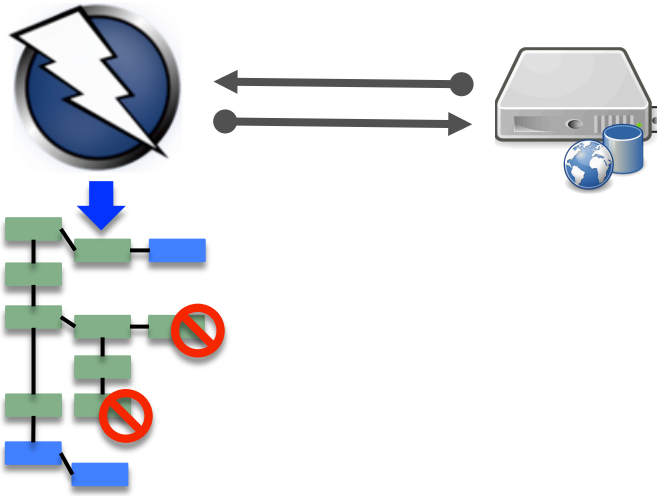


CARTOGRAPHIE PILOTÉE

2 – l'utilisateur **exclut** des liens dangereux

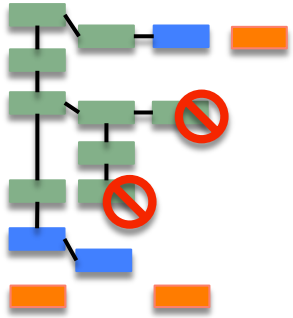


3 – on **complète** la cartographie à l'aide du mode automatique



RECONNAISSANCE

Recherche de contenu caché



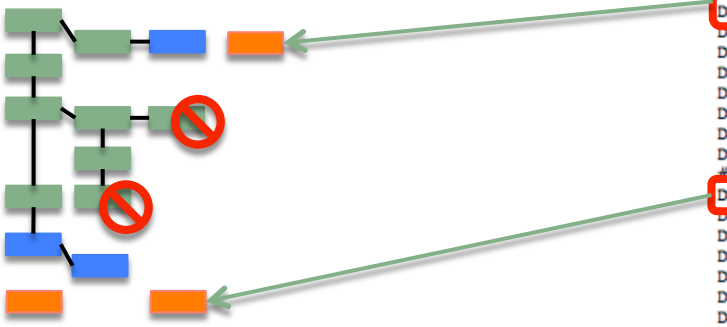
- liens cachés (parties réservées de l'application) ;
- code de test ;
- fichiers de backup ;
- fichiers de logs ;
- etc…

RECHERCHE DE CONTENU CACHÉ

- **robots.txt**

```
User-agent: *
Disallow: /admin/
Disallow: /awstats/
```

Interface d'administration
Outil de statistiques
…



www.[redacted]/robots.txt

Disable ▾ Cookies ▾ CSS ▾ Forms ▾ Images ▾ Informat

```
#
# robots.txt
#
# This file is to prevent the crawling and indexing of certain parts
# of your site by web crawlers and spiders run by sites like Yahoo!
# and Google. By telling these "robots" where not to go on your site,
# you save bandwidth and server resources.
#
# This file will be ignored unless it is at the root of your host:
# Used:    http://example.com/robots.txt
# Ignored: http://example.com/site/robots.txt
#
# For more information about the robots.txt standard, see:
# http://www.robotstxt.org/wc/robots.html
#
# For syntax checking, see:
# http://www.sxw.org.uk/computing/robots/check.html

User-agent: *
Crawl-delay: 10
# Directories
Disallow: /includes/
Disallow: /misc/
Disallow: /modules/
Disallow: /profiles/
Disallow: /scripts/
Disallow: /themes/
# Files
Disallow: /CHANGELOG.txt
Disallow: /cron.php
Disallow: /INSTALL.mysql.txt
Disallow: /INSTALL.pgsql.txt
Disallow: /install.php
Disallow: /INSTALL.txt
Disallow: /LICENSE.txt
Disallow: /MAINTAINERS.txt
Disallow: /update.php
Disallow: /UPGRADE.txt
Disallow: /xmlrpc.php
# Paths (clean URLs)
Disallow: /admin/
Disallow: /comment/reply/
Disallow: /filter/tips/
Disallow: /logout/
Disallow: /node/add/
Disallow: /search/
Disallow: /user/register/
Disallow: /user/password/
Disallow: /user/login/
# Paths (no clean URLs)
Disallow: /?q=admin/
```

RECHERCHE DE CONTENU CACHÉ

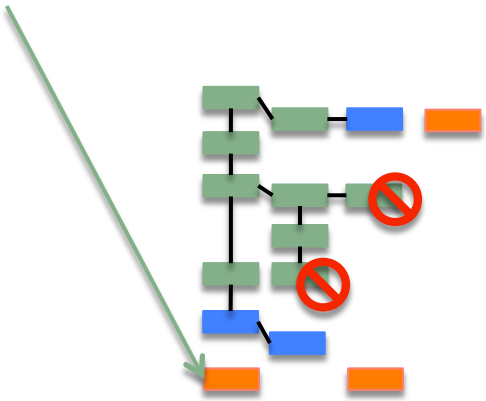
Force brute : listes de noms de fichiers/ répertoires (Dirbuster).

- « Parcours forcés » de ZAP
- Burp Intruder de BurpSuite.

Attention aux boucles infinies si l'application WEB renvoie un 200 OK + page d'erreur standard…

RECHERCHE DE CONTENU CACHÉ

- revue du code client (html, javascript) : commentaires développeurs…
- raisonnement par inférence :
  - RapportActivites2014.pdf => RapportActivites2013.pdf
  - ForgotPassword.php => ResetPassword.php



DÉMO : CARTOGRAPHIE ZAP

RECONNAISSANCE

OSINT (OpenSource Intelligence)

- Principe : utiliser les sources publiques
- Avantage : totalement furtif

Ex : Google hacking

OSINT : GOOGLE HACKING

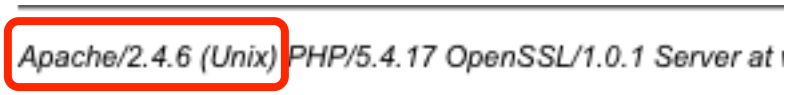
Quelques opérateurs de recherche avancée

Opérateur	Rechercher
intitle:*mot*	le mot dans le titre (élément HTML title).
inurl:*mot*	le mot dans l'URL de la page
intext:*mot*	le mot dans le corps du texte (élément HTML body)
filetype:*type*	un type de document (xls, pdf, doc, …)
site:*domaine*	les pages indexées pour le domaine ex. site:www.cnrs.fr
" phrase "	la phrase exacte
*	un mot : *ex* "un * DVD" => un lecteur DVD, un graveur DVD
.	un caractère quelconque
-*mot*	les réponses qui n'ont pas ce mot ex -intitle:forum
+*mot*	le mot (force la prise en compte de : le, et, de, …)
AND, OR	recherche ET (par défaut) / OU entre les termes

OSINT : GOOGLE HACKING

Informations sur le serveur HTTP

GOOGLE intitle:"index of" intext:"Apache/2*" intext:"server at"

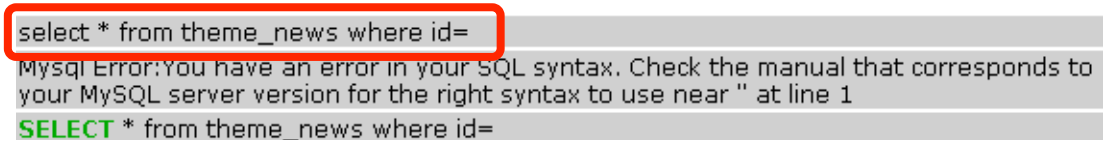


Recherche ciblée

GOOGLE site:www.ac-****** intitle:"index of" intext:"Microsoft.IIS/* server at"

Erreurs MySQL

GOOGLE mysql error -intitle:error -intitle:php -forum -forums -intitle:mysql

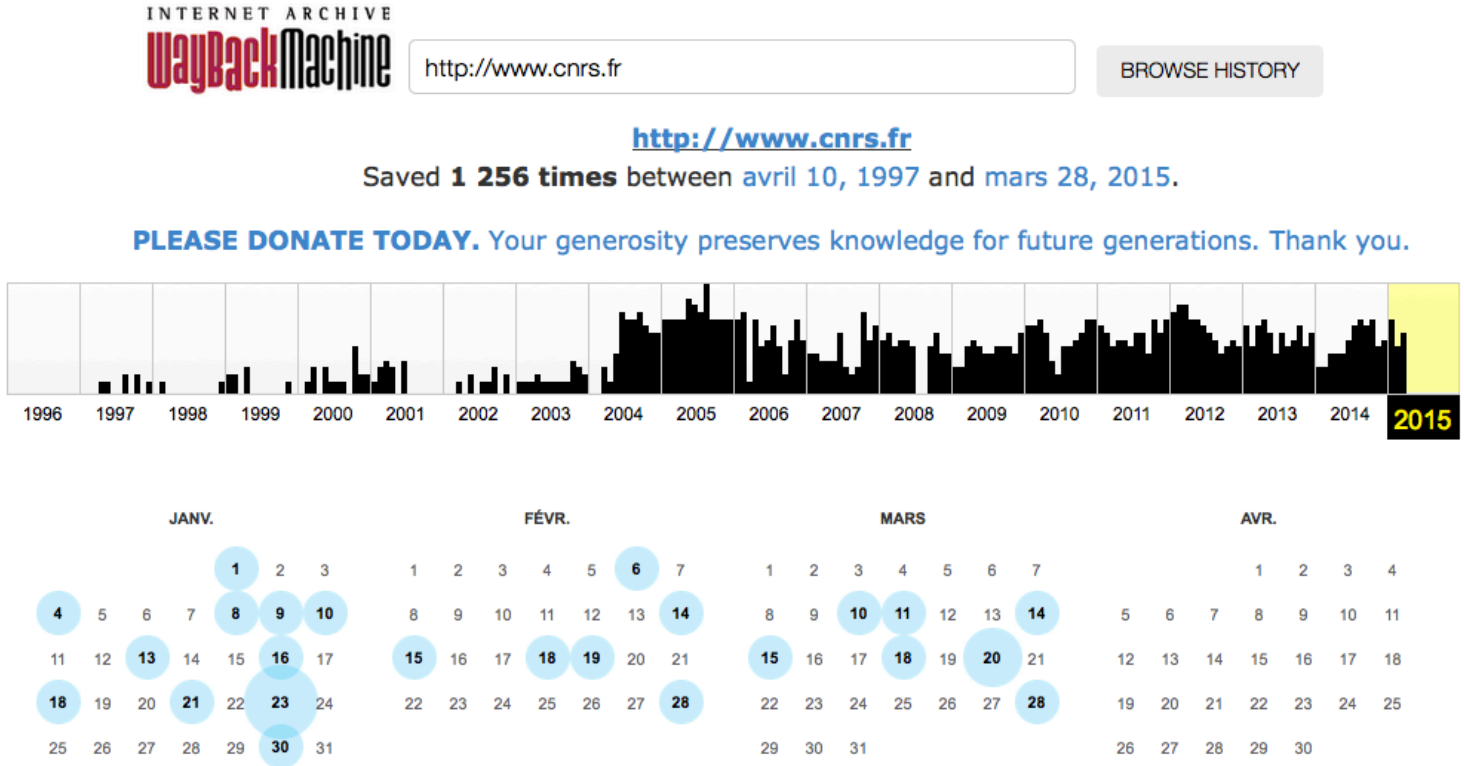


OSINT : GOOGLE HACKING DATABASE

En panne d'inspiration ? http://www.exploit-db.com/google-dorks/

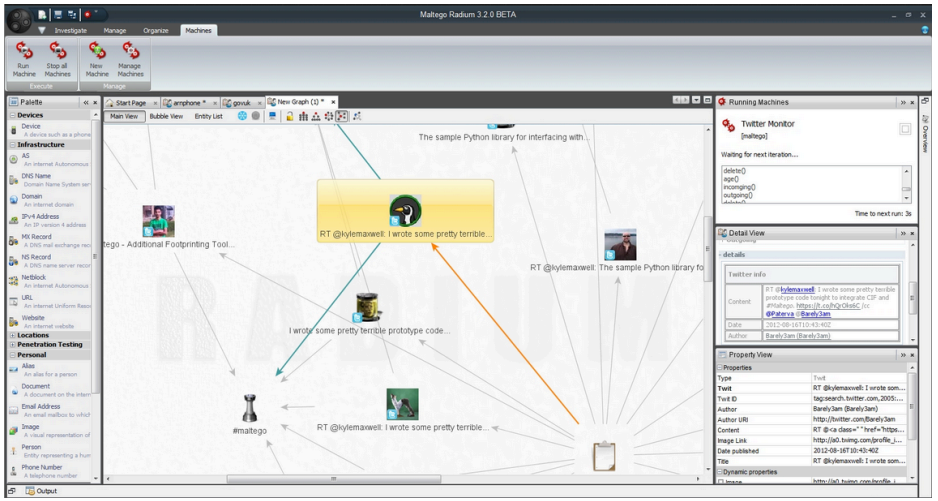
OSINT : RETOUR VERS LE PASSÉ

Retrouver des fonctionnalités passées/cachées : http://archive.org/web/



RECONNAISSANCE : OSINT

- identifier les développeurs de l'application ;
- leurs interventions sur les forums ;
- facebook, twitter… ;
- bouts de code ;
- analyser leurs habitudes ;



MALTEGO (https://www.paterva.com/web6/products/maltego.php)

RECONNAISSANCE : INFRASTRUCTURE

Failles au niveau infrastructure web :

- Type de serveur, version ou modules installés ;
- Système d'exploitation
- Défauts de configuration httpd ;
- Composants logiciels (CMS…) ;
- …

RECONNAISSANCE : INFRASTRUCTURE

Scanners de vulnérabilités

- WAPITI 2.3.0(10/2013)
- NIKTO 2.1.5 (03/2015)
- ARACHNI 1.0.6 (12/2014)
- SKIPFISH 2.10b (12/2012)
- W3AF (03/2014)

RECONNAISSANCE : INFRASTRUCTURE

Les fonctionnalités :

- Identifier le système d'exploitation, serveur, langage de programmation ;
- Fichiers intéressants (phpinfo, passwd.txt…) ;
- Scripts d'upload ;
- recherche de vulnérabilités (injections de code, SQL, LDAP, XSS, XPATH, CSRF, etc… indexes de répertoires, CMS obsolètes)

RECONNAISSANCE : INFRASTRUCTURE

Les limites :

Génèrent de nombreux faux positifs/négatifs

⇒ Nécessite de vérifier manuellement les résultats.

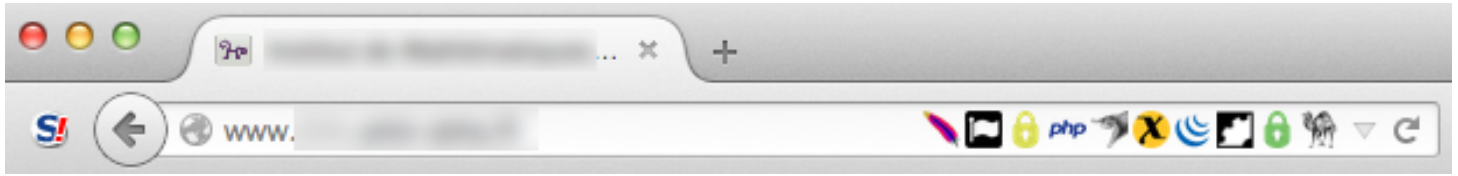
Attention au paramétrage de l'outil (scans hors contexte…, boucles infinies – page d'erreur unifiée)

DEMO : NIKTO

RECONNAISSANCE : INFRASTRUCTURE

WAPPALYZER
IDENTIFIES SOFTWARE ON THE WEB

Extension Firefox/Chrome



Identification des technologies web

Apache 2.2.20
Serveur web

Font Awesome
Script de police

OpenSSL 1.0.0e
Extension de serveur web

PHP 5.2.17
Language de programmation

SPIP
CMS

UNIX
Système d'exploitation

jQuery
Framework JavaScript

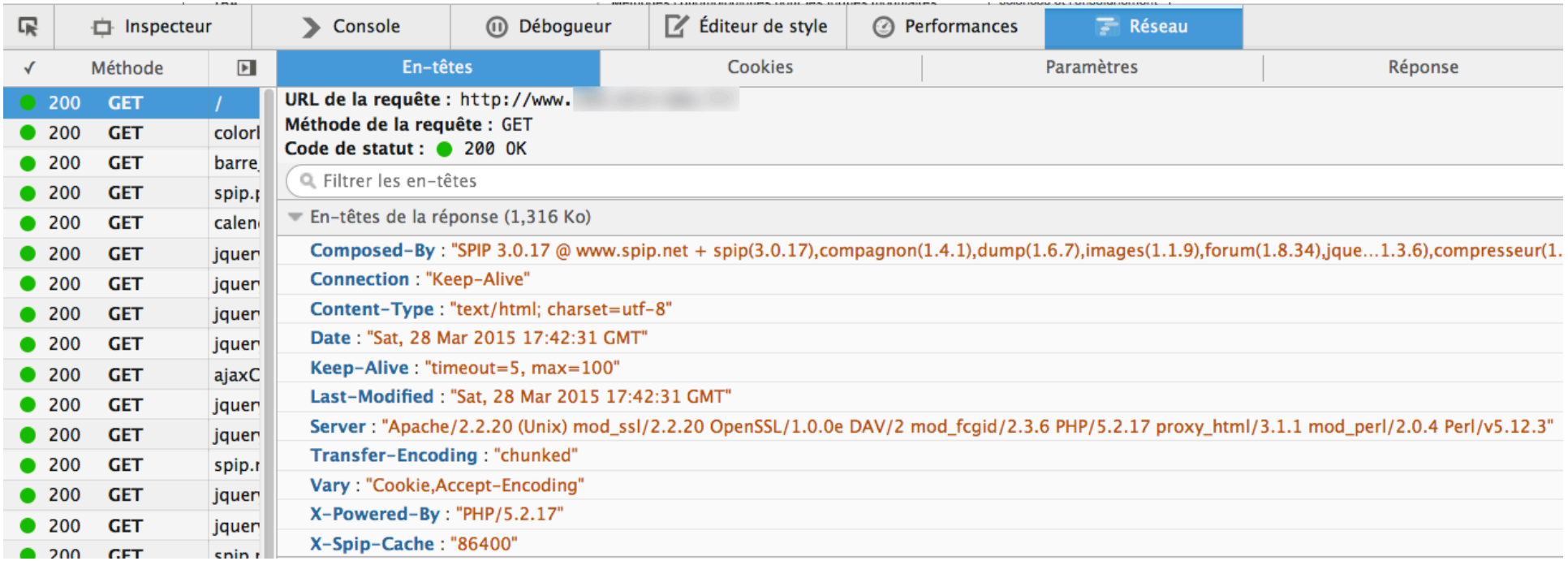
mod_perl 2.0.4
Extension de serveur web

mod_ssl 2.2.20
Extension de serveur web

Perl
Language de programmation

RECONNAISSANCE : INFRASTRUCTURE

Analyse de l'en-tête de la réponse HTTP (inspecteur du navigateur) :



RECONNAISSANCE : INFRASTRUCTURE

http://toolbar.netcraft.com/site_report?url=http://...

Site report for www.

Search...

Lookup another URL:

Share:

Enter a URL here

Netcraft Extension

- Home
- Download Now!
- Report a Phish
- Site Report
- Top Reporters
- Incentives for reporters
- Phishiest TLDs
- Phishiest Countries
- Phishiest Hosters
- Phishing Map
- Takedown Map
- Most Popular Websites
- Branded Extensions
- Tell a Friend

Phishing & Fraud

- Phishing Site Feed
- Hosting Phishing Alerts
- SSL CA Phishing Alerts
- Registry Phishing Alerts
- Domain Registration Risk
- Bank Fraud Detection
- Phishing Site Countermeasures

Extension Support

- FAQ
- Glossary
- Contact Us
- Report a Bug

Tutorials

- Installing the Extension
- Using the Extension
- Getting the Most
- Reporting a Phish

Background

Network

Hosting History

Netblock owner	IP address	OS	Web server	Last seen (Refresh)
[redacted]	[redacted]	Linux	Apache/2.2.22 Ubuntu	16-Jul-2014
[redacted]	[redacted]	Linux	Apache	26-May-2014
[redacted]	[redacted]	Linux	Apache	20-Dec-2011
[redacted]	[redacted]	Linux	Apache/2.2.11 Debian mod_jk/1.2.20-dev PHP/5.2.6-3 with Suhosin-Patch mod_python/3.3.1 Python/2.5.4 mod_ssl/2.2.11 OpenSSL/0.9.8g	8-Jun-2009
[redacted]	[redacted]	Linux	Apache/2.2.11 Debian mod_jk/1.2.20-dev PHP/5.2.6-3 with Suhosin-Patch mod_python/3.3.1 Python/2.5.2 mod_ssl/2.2.11 OpenSSL/0.9.8g	7-Mar-2009
[redacted]	[redacted]	Linux	Apache/2.2.9 Debian mod_jk/1.2.20-dev PHP/5.2.6-1lenny2 with Suhosin-Patch mod_python/3.3.1 Python/2.5.2 mod_ssl/2.2.9 OpenSSL/0.9.8g	29-Jan-2009
[redacted]	[redacted]	Linux	Apache/2.2.9 Debian mod_jk/1.2.20-dev PHP/5.2.6-0.1lenny1 with Suhosin-Patch mod_python/3.3.1 Python/2.5.2 mod_ssl/2.2.9 OpenSSL/0.9.8g	27-Jan-2009
[redacted]	[redacted]	Linux	Apache/2.2.9 Debian mod_jk/1.2.20-dev PHP/5.2.6-5 with Suhosin-Patch mod_python/3.3.1 Python/2.5.2 mod_ssl/2.2.9 OpenSSL/0.9.8g	15-Oct-2008
[redacted]	[redacted]	Linux	Apache/2.2.9 Debian mod_jk/1.2.20-dev PHP/5.2.6-3 with Suhosin-Patch mod_python/3.3.1 Python/2.5.2 mod_ssl/2.2.9 OpenSSL/0.9.8g	22-Sep-2008
[redacted]	[redacted]	Linux	Apache/2.2.9 Debian mod_jk/1.2.20-dev PHP/5.2.6-2 with Suhosin-Patch mod_python/3.3.1 Python/2.5.2 mod_ssl/2.2.9 OpenSSL/0.9.8g	28-Jul-2008

Si aucune information dans la réponse HTTP => prise d'empreinte HTTP

RECONNAISSANCE : INFRASTRUCTURE

Différences HTTP :

- lexicales (ponctuation, phrases)

Requête	Apache/2.4.6	IIS/5.0
GET /**xxx** HTTP/1.1	404 Not Found	404 **Object** Not Found

- syntaxiques (ordre des champs, valeurs listées)

Requête	Apache/2.4.6	IIS/8.0
HEAD / HTTP/1.1	**Date:** Server:	Server: **Date:**

- sémantiques (interprétation différente)

Requête	Apache/2.4.6	IIS/8.0
GET / HTTP/**5.0**	**200** OK	**400** Bad Request
GOT / HTTP/1.1	**405** Method Not Allowed	**404** Not Found

RECONNAISSANCE : INFRASTRUCTURE

https://w3dt.net/tools/httprecon

HTTPRecon (Server Fingerprint)

Target:
Tests: 9 test cases
Scan:
Export:

Contents

1. Summary
2. Matches
3. Responses
4. Details

Summary ↑

An advanced web server fingerprinting for the host and port tcp/80 was done with 9 test cases at

This analysis was able to determine the target httpd service as Apache 2.2.3 with 65 fingerprint hits in the database.

List of Matches ↑

	Name	Hits	Match
1.	Apache 2.2.3	65	100%
2.	AOLserver 4.0.10	63	96.92%
3.	Apache 1.3.33	63	96.92%
4.	Apache 2.0.46	62	95.38%
5.	nginx 0.6.16	62	95.38%

Cet outil laisse des traces dans les logs du serveur web !

RECONNAISSANCE : INFRASTRUCTURE

Identification des technologies côté serveur

- extensions de fichiers ;
- noms de répertoires ;
- jetons de session (JSESSIONID, ASPSESSIONID, PHPSESSIONID, etc…)

=> Recherche de vulnérabilités publiées pour tous les composants identifiés

RECONNAISSANCE : ANALYSE LOGIQUE

Analyser la logique de l'application

- Recenser les fonctionnalités :
  - principales, mais aussi secondaires
  - fonctions d'aide
  - génération de logs
  - gestion des erreurs
  - …
- Trouver des paramètres (GET/POST) « secrets » :

test/debug = 1/true/yes

RECONNAISSANCE : ANALYSE LOGIQUE

Recenser les mécanismes de sécurité :

- gestion des sessions,
- authentification,
- contrôle d'accès,
- gestion des mots de passe : changement, recouvrement…

RECONNAISSANCE : POINTS D'ENTRÉES

Inventorier **tous les points d'entrées** utilisateurs (y compris cachés)
+ les traitements réalisés sur les données utilisateur :

- URL
- Paramètres query string
- Paramètres POST
- Cookies
- En-têtes HTTP

=> Vecteurs potentiels d'attaques

RECONNAISSANCE : VECTEURS PÉRIPHÉRIQUES

Identifier les vecteurs d'attaque périphériques.

Données échangées avec d'autres serveurs :

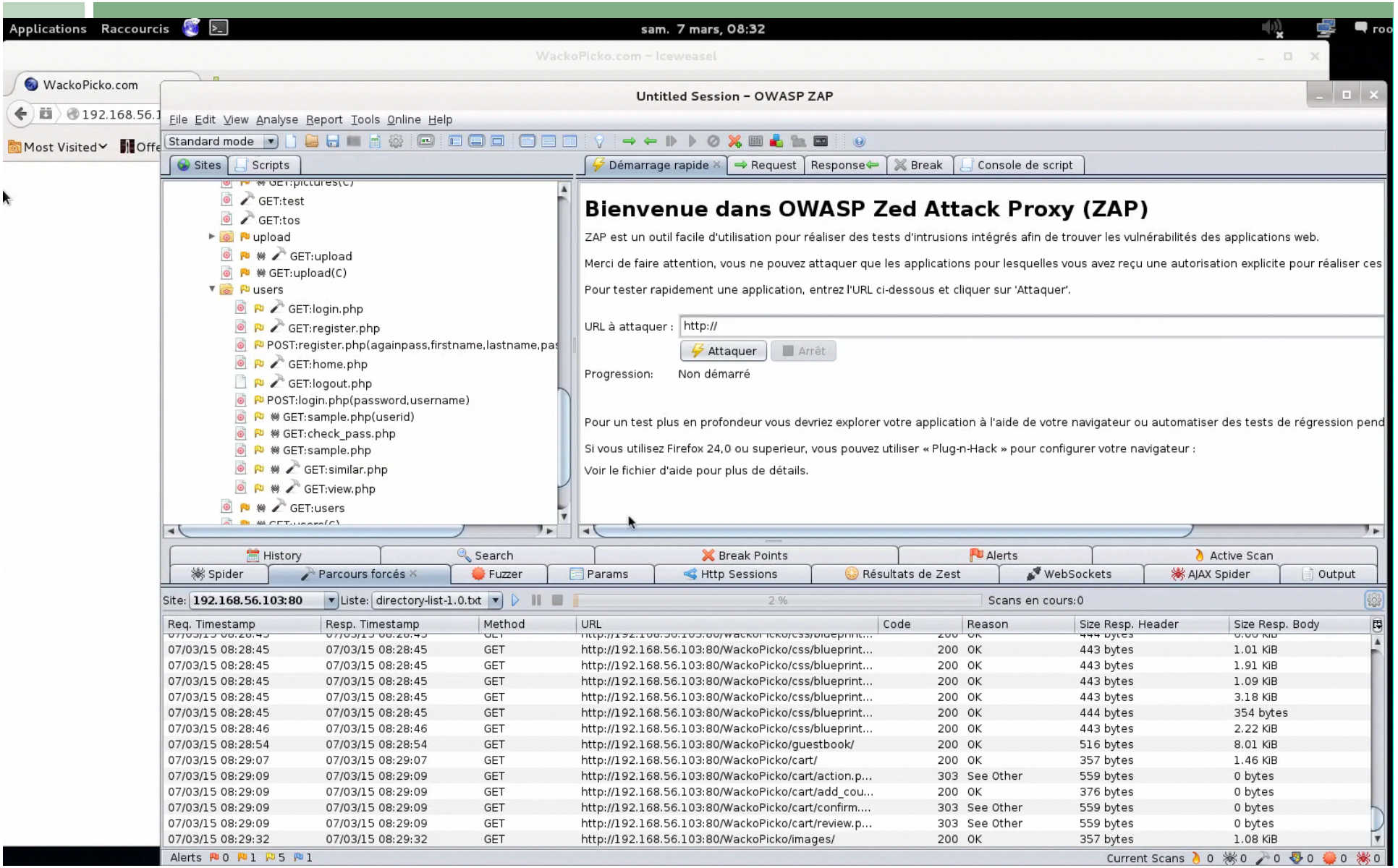
- Serveur de mails
- Autres serveurs webs
- Webservices
- NIDS analyse le trafic ?

CARTOGRAPHIE DE LA SURFACE D'ATTAQUE

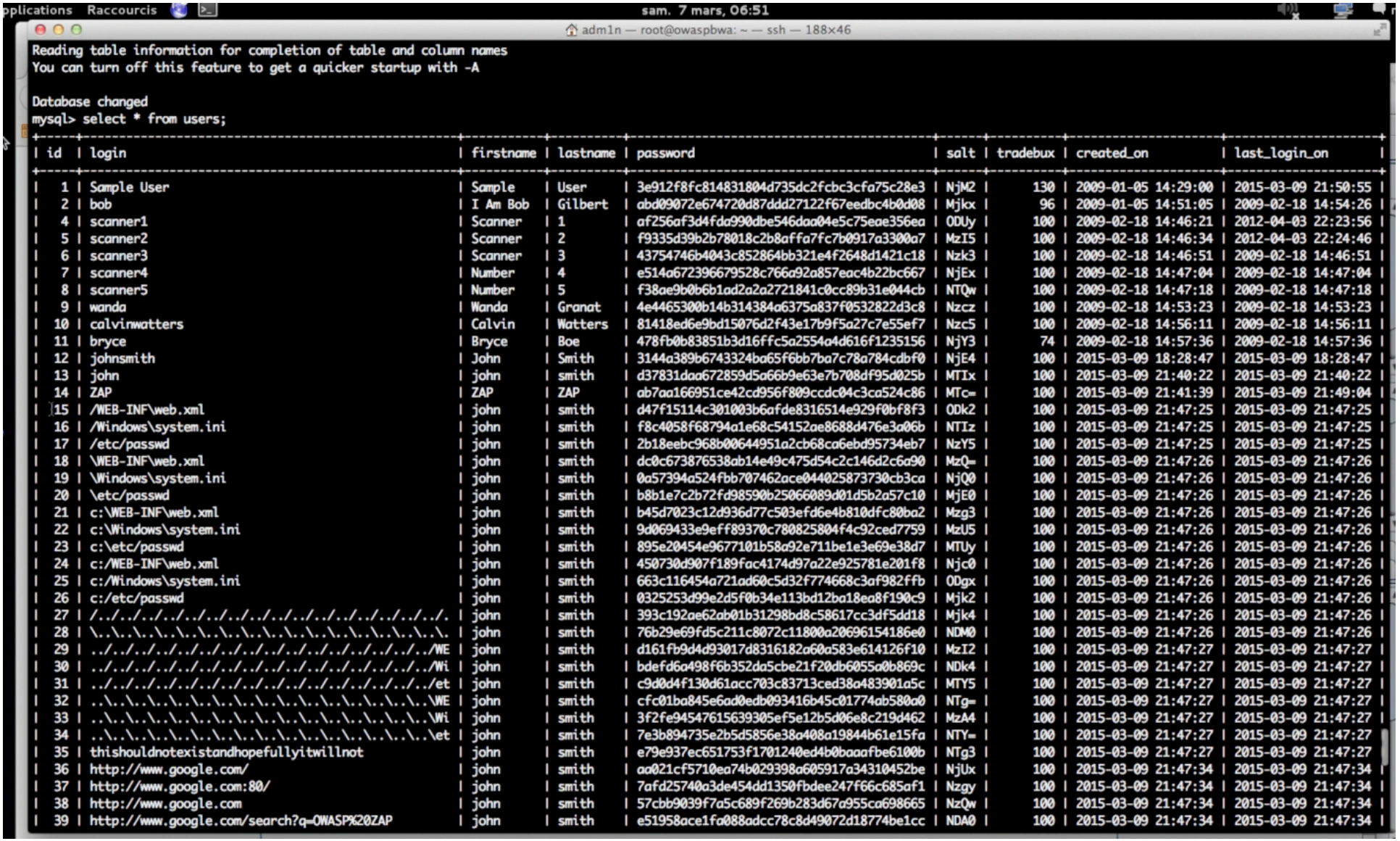
Associer vecteurs ⇔ types de vulnérabilités

- Interaction avec un SGBD = injection SQL
- Affichage d'information utilisateur = XSS
- Formulaire d'authentification = attaque de force brute, énumération de noms
- Upload de fichiers = traversée de chemin, dépôt de scripts
- Contrôle d'accès = élévation de privilèges
- Messages d'erreur (php, mysql) = fuites d'informations
- etc…

DÉMO : ZAP, SCAN ACTIF



DÉMO : ZAP, SCAN ACTIF : les dégâts…



EXPLOITATION

Exploitation : Attaque.

- **Altération méthodique** des données
  - Ajout de variables
  - Suppression de variables
  - Modification de valeurs de tous les paramètres recensés.

EXPLOITATION

- **But :**
  - Provoquer un changement de comportement de l'application
  - Révélateur d'une vulnérabilité exploitable :
    - Affichage de messages d'erreur (erreur php, mysql, java…)
    - Comportement spécifique au type de vulnérabilité testée (ex. fenêtre alert("XSS");)
    - Comportement anormal de l'application (attaque logique).

EXPLOITATION : Injections SQL

#1 du TOP 10 de l'OWASP 2013