Charles Olivier-Anclin
charles.olivier-anclin@uca.fr
perso.isima.fr/~chaolivi

Clermont Auvergne INP, Université Clermont Auvergne

ZZ2 F5 - Sécurité web

Introduction et mise en place

Organisation du cours - 15 sceances

Partie - 1 Contexte

1. **Aujourd'hui :** mise en place et découverte des outils.
2. Base de la sécurité et menaces.
3. Introduction à la cryptographie et au respect de la vie privé.
4. Architecture des applications Web
   - HTTP
   - Browser & cookies
   - Javascript
   - Certificats
   - HTTPS et ces vulnérabilités.
     - Heartbleed
5. Authentification
   - Brute force
   - Session

Partie 2 - Vulnérabilités communes

1. Injection de commandes
   - Shellshock
2. Injection de fichiers
3. Upload
4. XSS
   - CSP
   - SOP/CORS
5. LFI et RFI
6. CSRF
7. SQLi
   - Drupalgeddon

Point légal

Les outils présentés ici doivent être utilisés **uniquement dans un cadre autorisé**. Tout usage sur un réseau tiers sans autorisation explicite, est **formellement interdit**.

> ***Article 323-1 du Code pénal :*** *Le fait d'accéder ou de se maintenir, frauduleusement, dans tout ou partie d'un système de traitement automatisé de données est puni de* ***2 ans d'emprisonnement*** *et de* ***60 000 Euro d'amende****.*
> *Si cette intrusion entraîne : la suppression ou modification de données, ou une altération du fonctionnement du système, la peine peut aller jusqu'à* ***3 ans d'emprisonnement*** *et* ***100 000 Euro d'amende****.*
>
> Legifrance

Aller plus loin : Articles 323-2,3,4,7

Cours en construction

Cours précédement donnée par Vincent Mazenod

- Première année que j'enseigne ce cours.
- Votre feedback est le bienvenu à tout instant !
  - Erreurs
  - Manques
  - Choses superflues
  - ...
- Posez des questions, intéragissez.

Examen (10 points)

Écrit en fin de session - Durée : 1h00 - Support : interdit

Présentation de groupe au début de cours (8 points)

Durée : 15 minutes. Présenter l'une des catgories de vulnérabilité du OWASP Top 10:2025 :

- présentation des risques
- proposition et démonstration d'un scénrio d'attaque
- DVWA est interdit car le cours le traite déjà
- proposition de mesure à prendre

Après la présentation vous m'enverrez par mail

- le support de présentation (ppt, url, ...)
- l'url de l'exploit et/ou le code nécessaire à l'exploitation (zip/ou repo git)

Hedgedoc pour la répartition des groupes

Note technique (2 points)

- l'url `http://vm-<username>.local.isima.fr` devra renvoyer un code HTTP 200
- l'url `http://vm-<username>.local.isima.fr/websec` devra renvoyer un code HTTP 401
- l'url `http://kali:kali@vm-<username>.local.isima.fr/websec` devra renvoyer un code HTTP 200

vérifiez à tout moment que vous aurez le maximum de points avec les websec-checker [1].

1. https://gitlab.isima.fr/vimazeno/websec-checker.git

Outils en cours

1. Manipulation sur le navigateur (cookies, requettes)
2. Burp
3. Wireshark
4. DVWA instalation
5. Autres possibilités

- http://proxy - la passerelle percée
- http://debian - la mututalisée moisie
- http://debian11 - la fresh
- http://thenetwork - https://www.digitalocean.com/community/tutorials/how-to-use-sshfs-to-mount-remote-file-systems-over-ssh

Virtual Box

Obtenir kali : https://www.kali.org/get-kali/

Survival cheatsheet Un tutoriel

Les Machines virtuels

Vm offensive : Kali https://www.kali.org/.

- C'est à partir de cette VM que vous suivrez le cours et lancerez toutes les attaques
- nom d'utilisateur : kali – mot de passe : kali

La cible : DVWA, http://dv.wa

- C'est la VM qui héberge l'application vulnérable.
- nom d'utilisateur DVWA : admin – mot de passe DVWA : password

VM debian11 : est une fresh install de debian 11, qui vous permettra d'installer DVWA

- nom d'utilisateur : kali – mot de passe : kali

VM proxy : héberge la vulnérabilité heratbleed présenté en début de cours

- nom d'utilisateur : kali – mot de passe : kali

VM debian : héberge la vulnérabilité drupalgeddon présentée en fin de cycle de cours

- nom d'utilisateur : kali – mot de passe : kali

Recréer l'environnement de cours

Télécharger les images OVA : https://owncloud.isima.fr/s/uWs5lAn0FdULdt4

- kali.ova ; dvwa.ova ; debian.ova ; proxy.ova ; debian11

Créer un réseau NAT

```
vboxmanage  natnetwork add --netname natwebsec --
network "172.16.76.0/24" --enable --dhcp off
```

Importer les images OVA

```
vboxmanage import nom_d'une_vm.ova
```

Configurer le réseau pour chaque vm

```
vboxmanage modifyvm nom_d'une_vm --nic1 natnetwork --nat-
network1 natwebsec
```

(Debug) En cas de réseau injoignable sur proxy

`ping 172.16.76.145  # ping sur kali` 

- retourne `connect: Network is unreachable`

Vérifier le numéro de votre interface réseau

```
student@proxy:~$ ifconfig -a
eth2      Link encap:Ethernet  HWaddr 08:00:27:ae:b5:20
....
```

par exemple ce numéro peut être **eth2 au lieu de eth0**. Il faut alors **modifier le fichier** `/etc/network/interfaces` en fonction.

```
student@proxy:~$ sudo vi /etc/network/interfaces
....
# The primary network interface
auto eth2
iface eth2 inet static
....
```

puis activer l'interface réseau `sudo ifup eth2`
Ce bug est dû à la numérotation fantaisiste d'Ubuntu des interfaces réseau ...

Liste des vms / noms de domaine (/etc/hosts)

```
# SecLab

## proxy
172.16.76.143 proxy secured heart.bleed fo.ol #proxied version of dum.my

# debian
172.16.76.144 debian good.one go.od targ.et mutillid.ae d.oc dum.my spip sp.ip
172.16.76.144 drup.al hackable-drupal.com drupal wordpre.ss bl.og wp wordpress

# kali
172.16.76.145 kali bad.guy hack.er 1337.net

# dvwa
172.16.76.146 dvwa dvwa.com dv.wa

# debian11
172.16.76.147 debian11

# host
172.16.76.1   us.er
```

Section 1 : **La sécurité Web - Introduction**

Pourquoi sécuriser notre utilisation du numérique ?

Bien d'autres...

Le site de l'ANSSI « a pour objectif de répondre à vos questions en matière de cybersécurité et de partager avec vous des informations ciblées et accessibles. » [2]

Sécuriser un site web

Recommandations pour la mise en œuvre d'un site web : Maîtriser les standards de sécurité côté navigateur

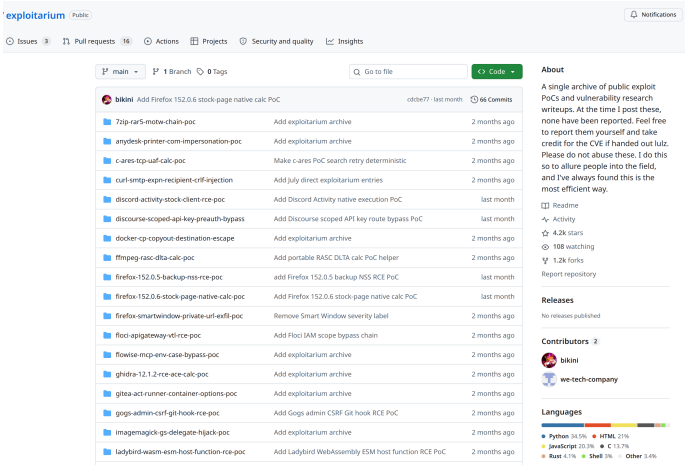
... et pleins d'autres !!

2. source : https://www.ssi.gouv.fr/

La cybersécurité aujourd'hui

- **241 jours** en moyenne pour identifier et contenir une fuite de données. [IBM, 2025]
- Les PME restent une **cible majeure** des cyberattaques. [Verizon DBIR, 2026]
- Principaux vecteurs d'accès initial : **exploitation de vulnérabilités (22 %)**, phishing (20 %) et compromission d'identifiants (15 %). [Verizon DBIR, 2026]
- Les rançongiciels restent une **menace majeure**. [Verizon DBIR, 2026]
- **61 %** des fichiers malveillants envoyés par e-mail sont des HTML. [Check Point, 2025]
- L'ANSSI a recensé **1 366 incidents** en France en 2025. [ANSSI, 2026]
- Le coût mondial de la cybercriminalité est estimé à **10 500 Md $/an**. [Cybersecurity Ventures]
- Les dépenses mondiales en sécurité de l'information devraient atteindre **244 Md$ en 2026**. [Gartner, 2026]
- En France, **144 compromissions par rançongiciel** recensées en 2024. [ANSSI, 2025]

130 exploits 0-day par IA



VLC - 7-Zip - Firefox - Gitea - ... (20ème de cibles)

“Stresser le code”

Fonctionnement

Test automatisé par injection de données aléatoires et parfois structurées.

But & Usage : Barton Miller, 1988 

- Générer un plantage
- Identification de vulnérabilité potentielles (*e.g.* injection SQL)
- Validation d’analyse statiques

Types de fuzzing

Selon la génération des entrées

- **Mutation-based** : modifie des données valides
- **Dumb fuzzing** : entrées aléatoires
- **Fuzzing grammatical** : respecte de la structure
- **Generation-based** : génère des données nouvelles

Selon la connaissance du système

- **Black-box** : aucune connaissance interne
- **Grey-box** : informations partielles
- **White-box** : accès au code

Techniques spécialisées

- **Coverage-guided** : explore de nouveaux chemins
- **Protocol fuzzing** : teste les protocoles réseau
- **File-format fuzzing** : teste les formats de fichiers
- **API fuzzing** : teste les paramètres d'API
- **Differential fuzzing** : compare plusieurs implémentations

Quelques outils sur l'étagère

Boîte grise :

AFL++, **libFuzzer**

feedback de couverture / enrichissement de code automatique

Boîte blanche :

SymCC, **LibFuzzer**

génération de contraintes + exécution symbolique

Fuzzing navigateur : **ClusterFuzz**, **Domato**, **Dharma**

JS, DOM, formats et APIs Web

Aucun outil n'est universel : le choix dépend de la cible, de l'instrumentation et du budget d'exécution.

Exercice - Pratiquer le fuzzing

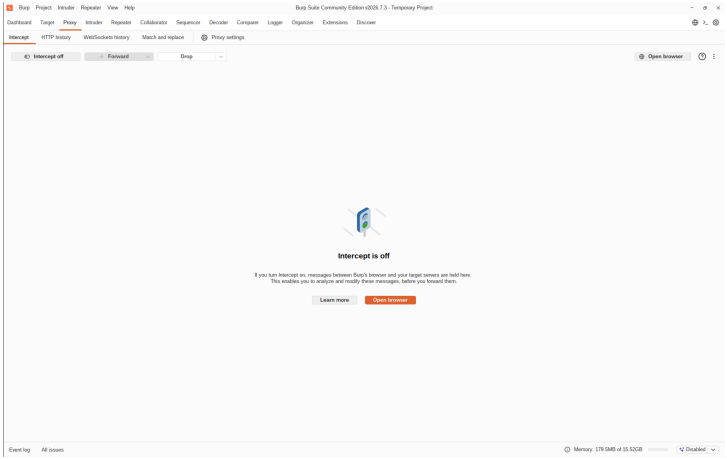
L'ensemble des éléments de l'activité sont à télécharger sur :

```
wget https://perso.isima.fr/~chaolivi/zz2f5-WebSec/1.fuzzing/
```

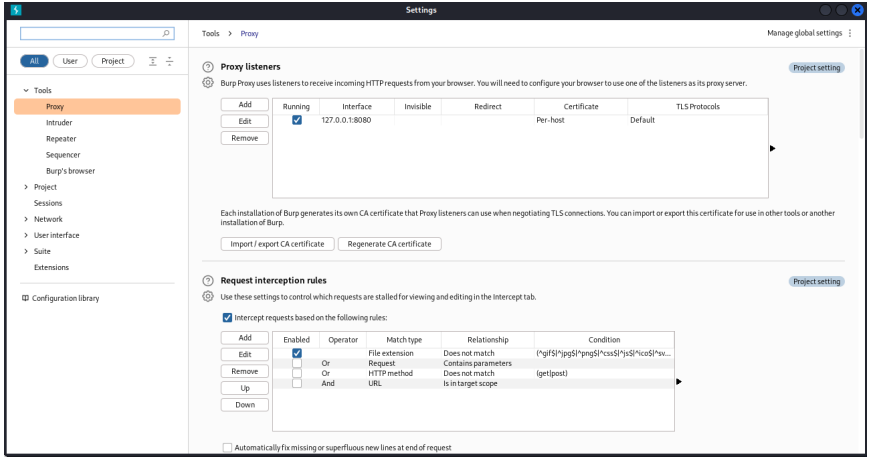
Exercice à réaliser en local.

Burp suite

- next / next / accept / ...
- Proxy

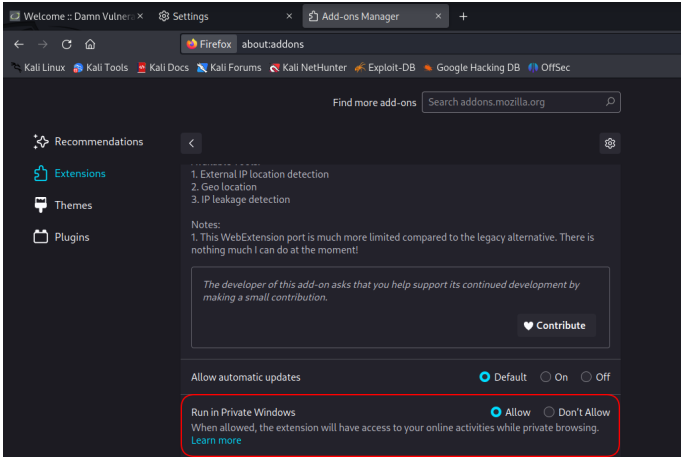


- next / next / accept / ...
- Proxy -> proxy settings



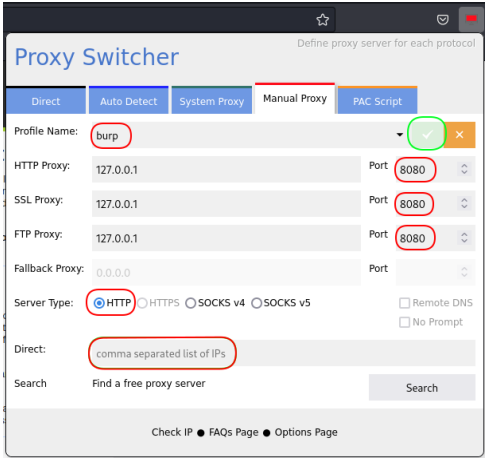
Toggle FF proxy

Installer Proxy Switcher and Manager



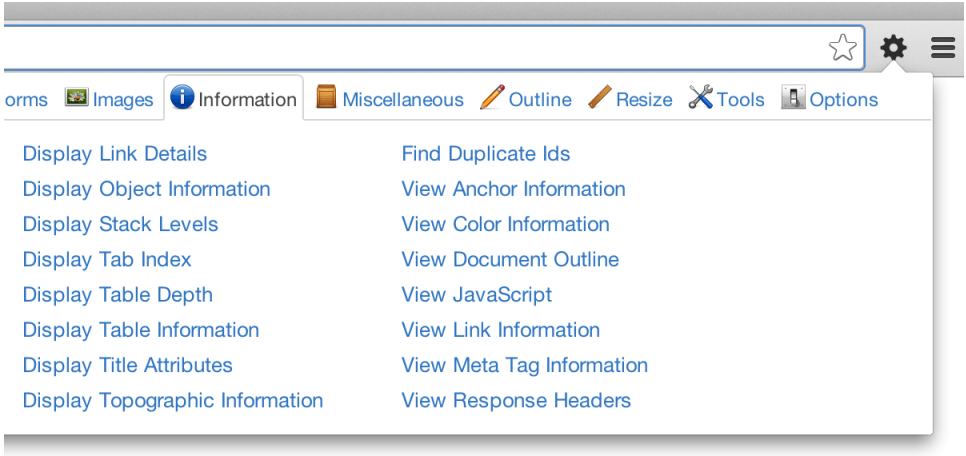
Toggle FF proxy

Installer Proxy Switcher and Manager



Web developer addons

Installer Web developer



Résumé

Organisation du cours :

Examen + présentation + note technique

Pourquoi la sécurité ?

Pour notre vie privée

Présentation des outils :

VM, DVWA, Burp, ...

Fuzzing :

testes et recherche de vulnérabilités

La suite : vie privée et cryptographie

Vie privée

- « Je n'ai rien à cacher »
- Cookies et tracking
- Fingerprinting
- Données personnelles
- Souveraineté numérique

Cryptographie

- Confidentialité et intégrité
- Authentification
- Chiffrement de bout en bout
- Tor et anonymat
- Les pièges des implémentations