Sommaire Cours

I. Introduction – problématique de la sécurité – sécurité web
II. Présentation des menaces – vulnérabilités
III. Standards de sécurité utiles pour le web
IV. Architecture du web (infrastructure et applicatif)
V. Sécurisation des services
VI. Sécurisation des applications
VII. Sécurisation de l'infrastructure

Classification et regroupement des menaces coté serveur et coté client

1. Injection de commandes
    a. Débordement de tampon (buffer overflow)
    b. Chaîne de format (format string)
    c. Injection LDAP
    d. Injection de commandes (OS Commanding)
    e. Injection SQL
    f. Injection SSI
    g. Injection XPath

2: **Révélation d'informations**
    a. Listing de répertoires (directory indexing)
    b. Fuite d'informations (information leakage)
    c. Traversée de chemin (path traversal)
    d. Prédiction de localisation de ressources (predictable resource location)

3: Authentification
    a. Force brute (brute force)
    b. Authentification insuffisante (insufficient authentication)
    c. Mauvais traitement des recouvrements de mot de passe (weak password recovery validation)

4: Autorisation
    a. Prédiction de session (credential/session prediction)
    b. Autorisation insuffisante (insufficient authaurization)
    c. Expiration de session insuffisante (insufficient session expiration)
    d. Fixation d'identifiant de session (session fixation)

5. Attaques côté client
    a. Usurpation de contenu (content spoofing)
    b. XSS (Cross Site Scripting)

6. Logiques
    a. Abus de fonctionnalité (abuse of functionality)
    b. Déni de service (denial of service)
    c. Anti-automatisation insuffisante (insufficient anti-automation)
    d. Validation insuffisante du flux logique de l'application

7. Autres
    a. HTTP Response Splitting / CR LF Injection
    b. Prise d'empreinte (Web Server/Application Fingerprinting)

* http://projects.webappsec.org/f/WASC-TC-v2_0.pdf

2:Révélation d'informations

a) Listing de répertoires (directory indexing) :
   - Possible si pas de fichiers html dans répertoire web ou par les répertoires indexés par google
   - Permet de voir les fichiers temporaires, fichiers cachés, de config, d'info,…

b) Traversée de chemin (path traversal)
   - Consiste a accéder à des fichiers ou répertoires qui seraient interdits d'accès si on les demandait directement (script PHP d'un autre utilisateur, .htaccess, …).
   - Ces attaques utilisent des **adresses relatives**

c) Fuite d'informations (information leakage)
   - Informations sur les logiciels , leurs versions
   - Données d'accès au système (fichiers de mot de passe)
   - Données personnelles (règlementation RGPD)

d) Prédiction de localisation de ressources (predictable resource location)
   - Attaque brute force pour obtenir des ressources dans des chemins standards

2: Révélation d'informations

Attaque en reconnaissance

- **Type d'attaque** : attaque de reconnaissance
- **Mécanismes** : consultation des informations publiques accessibles notamment via,
  - ✓ les bases whois ;
  - ✓ les moteurs de recherche.
- **Objectif** :
  - obtenir des informations nécessaires à la réalisation d'une intrusion ;
  - plus grave, obtenir des informations sensibles malencontreusement publiées sur Internet.
- **Caractéristiques** :
  - une grande partie de l'information est disponible sur Internet (Google haking) ;
  - utilisées par les pirates pour répertorier les sites sensibles.
- **Parades** :
  - sensibilisation des utilisateurs ;
  - utilisation d'outils dédiés pour éviter le fuite d'informations ;

2: Révélation d'informations

Attaque en reconnaissance : Google hacking

- **Recueil d'informations** :
  - **intitle:hacking** : recherche les pages web contenant le mot « hacking » dans leur titre (144 000 résultats),
  - **inurl:login** : recherche les pages contenant l'occurrence « login » dans leur url (3 710 000 résultats),
  - **intext:"md5 reverse hash"** : recherche les pages contenant la phrase « md5 reverse hash » dans leur corps (1 720 résultats),
  - **link:"www.blackhat.com"** : recherche les pages web contenant un lien vers www.blackhat.com (6 670 000 résultats),
  - **filetype:log :** recherche les fichiers dont le type ou l'extension est « log » (3 630 000 résultats),
- **Exploitation mise en cache de pages WEB**

2: Révélation d'informations

Attaque en reconnaissance : Google hacking

- **Parcours de répertoire(s) et fichier(s)** :
  - **intitle:"index of" :**De nombreux serveurs web affichent le contenu d'un répertoire de données en affichant en premier lieu « index of » suivi du chemin complet dans la barre de titre de la page web.
  - Affinage : « intitle:"index of" » permet de rechercher l'ensemble des répertoires visités par Google ayant le mot « admin » dans leur url.
  - Il est alors possible de remonter et d'explorer les différents répertoires en utilisant les liens « Parent Directory », etc…
  - La requête « intitle:"index of" router.cfg » permet de rechercher des répertoires contenant un fichier nommé « router.cfg ».
- **Adresses emails** :
  - **filetype: «From»** avec des mots-clés contenus dans les en-têtes SMTP (From, Subject, Received, Message- ID, …) « donne accès aux boites aux lettres disponibles sur Internet.

Sécurisation d'Apache

- Limiter les informations visibles
- Empêcher le parcours des répertoires et les liens symboliques
- Protéger les répertoires par mots de passe
- Limiter les DoS
- Sécuriser les authentifications

2a: **Listing de répertoires**

Empêcher le parcours et l'accès aux répertoires

- Droits sur les répertoires Apache

  - par défaut est installé/paramétré/démarré par l'utilisateur root

  - devient ensuite la propriété de l'utilisateur www-data
  - Les répertoires doivent être en 755 drwxr-xr-x
  - L'utilisateur ne doit pas posséder de privilèges qui lui permettraient d'accéder à des fichiers non destinés au monde extérieur
  - l'utilisateur ne doit pas exécuter de code dont l'usage soit destiné à un usage autre que les requêtes HTTP.

2a: **Listing de répertoires**

Empêcher l'accès aux répertoires

- Gestion pour le serveur complet et tout les sites web dans
  /etc/apache2/apache2.conf
  /etc/apache2/conf.available/security.conf

- Gestion **par site web** dans /etc/apache2/site-available/monsite.conf

- Ordre d'évaluation des droits d'accès apache2, security et siteweb.

2a: Listing de répertoires

Empêcher l'accès aux répertoires

- La syntaxe pour bloquer l'accès d'un répertoire par un domaine est la suivante :

Apache2.2 et précédents

```
Order (Allow,Deny ou Deny,Allow)
Allow (all, [liste de domaine])
Deny (all, [liste de domaine])
```

Apache2.4

```
Require all denied
Require all granted
```

- **Order** précise dans quel ordre les directives 'deny' et 'allow' sont évaluées.
- **Deny from all** interdit l'accès depuis "tout".
- Exemple d'accès au répertoire /var/www de apache
- Avec cette directive, vous n'accédez plus au Serveur Apache.
- Ensuite vous autorisez au cas par cas.**Allow from** `192.168.0.15`

```
<Directory /var/www>
        Order deny, allow
        Deny from all
        Allow from 192.168.0.15
</Directory>
```

```
<Directory /var/www>
        Require all denied
        Require ip 192.168.0.15/24
</Directory>
```

2b: Fuite d'informations

Empêcher le parcours des répertoires

Plusieurs options permettent le parcours de répertoires. Ces options doivent être supprimées

- Le suivi de lien symboliques (option FollowSymLimks)
- Le parcours des sous-répertoires, (option Indexes)

Apache2.2

```
<Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride All
        Order allow,deny
        allow from all
</Directory>
```

Apache2.4

```
<Directory /var/www/monsiteweb >
        Options FollowSymLinks
            AllowOverride None
        Require all denied
</Directory>
<Directory /var/www/monsiteweb >
        Options Indexes
        AllowOverride None
            Require all granted
</Directory>
```

2c: Traversée de chemin

Parcours via adresses relatives

Plusieurs options permettent le parcours de répertoires. Ces options doivent être supprimées

- Le suivi de lien symboliques (option FollowSymLimks)
- Le parcours des sous-répertoires, (option Indexes)

```
<Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride All
        Order allow,deny
        allow from all
</Directory>
```

2d: **Prédiction de localisation de ressources** Usage Fail2ban

- **fail2ban n'est pas un outil de sécurité**.
  - analyse les logs de divers services (SSH, Apache, FTP…)
  - cherche des correspondances entre des motifs définis dans ses filtres et les entrées des logs
  - Lorsqu'une correspondance est trouvée une ou plusieurs actions sont exécutées
  - **Il évite la surcharge des systèmes de logs en bloquant certaines adresses ou domaines**

- Failtoban : service à installer et à configurer qui analyse par service
- Installation : `apt-get instal fail2ban`
- log dans var/log/fail2ban.log

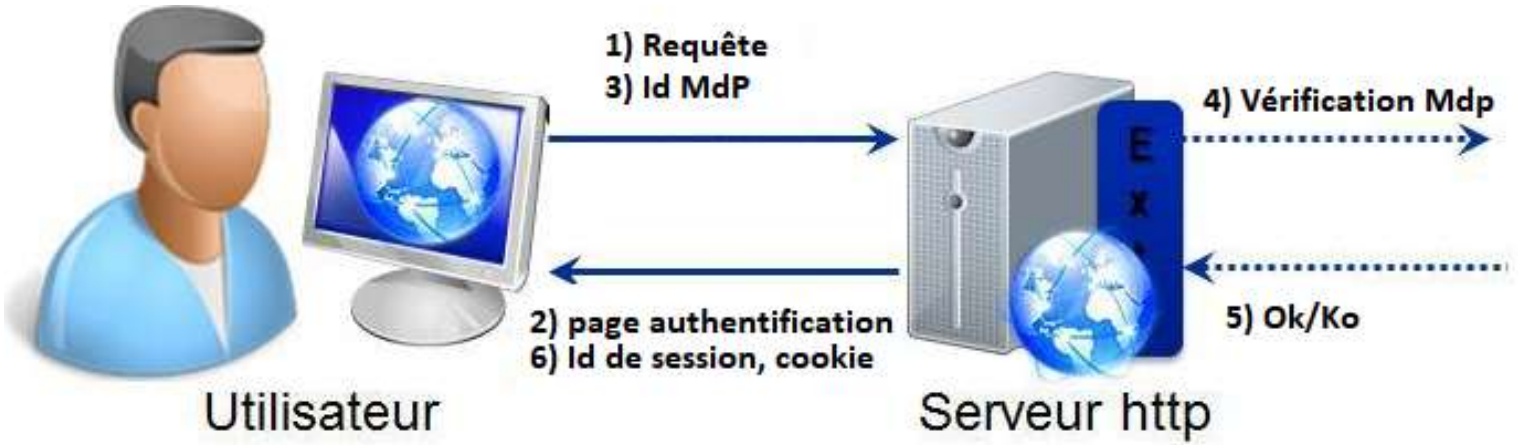
Sécurisation des services

1. Injection de commandes
    a. Débordement de tampon (buffer overflow)
    b. Chaîne de format (format string)
    c. Injection LDAP
    d. Injection de commandes (OS Commanding)
    e. Injection SQL
    f. Injection SSI
    g. Injection XPath
2: Révélation d'informations
    a. Listing de répertoires (directory indexing)
    b. Fuite d'informations (information leakage)
    c. Traversée de chemin (path traversal)
    d. Prédiction de localisation de ressources (predictable resource location)
3: Authentification
    a. Force brute (brute force)
    b. Authentification insuffisante (insufficient authentication)
    c. Mauvais traitement des recouvrements de mot de passe (weak password recovery validation)
4: Autorisation
    a. Prédiction de session (credential/session prediction)
    b. Autorisation insuffisante (insufficient authaurization)
    c. Expiration de session insuffisante (insufficient session expiration)
    d. Fixation d'identifiant de session (session fixation)
5. Attaques côté client
    a. Usurpation de contenu (content spoofing)
    b. XSS (Cross Site Scripting)
6. Logiques
    a. Abus de fonctionnalité (abuse of functionality)
    b. Déni de service (denial of service)
    c. Anti-automatisation insuffisante (insufficient anti-automation)
    d. Validation insuffisante du flux logique de l'application
7. Autres
    a. HTTP Response Splitting / CR LF Injection
    b. Prise d'empreinte (Web Server/Application Fingerprinting)

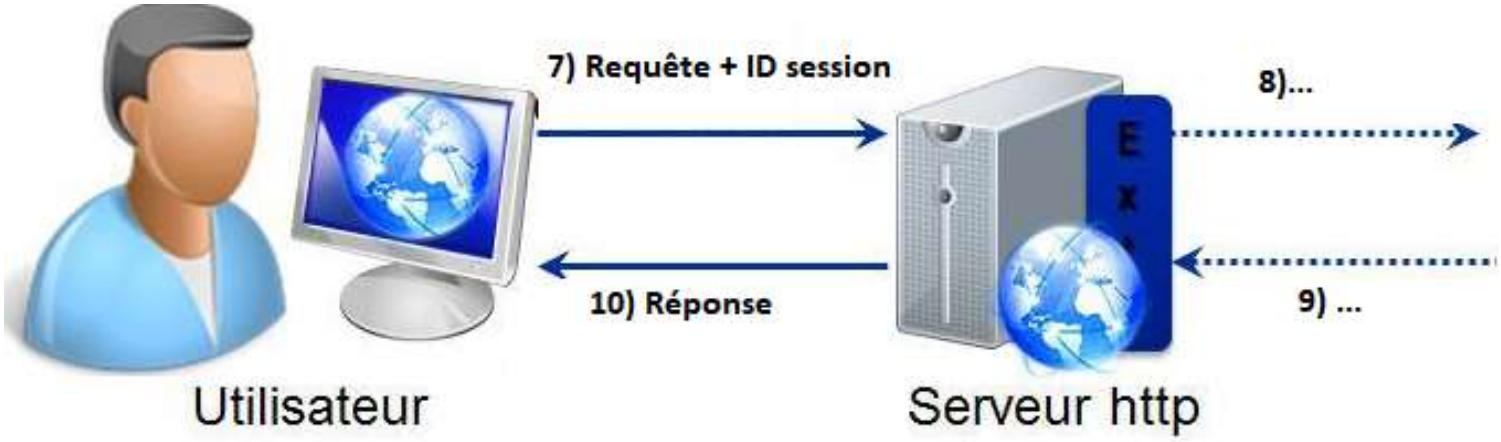
* http://projects.webappsec.org/f/WASC-TC-v2_0.pdf

3: Mécanisme d'authentification

- Etape 1 Authentification



- Etape 2 connexion avec Id de session



3: **Attaques en authentification**

a. Force brute :
   - essai/erreur sur ID/MDP (importance du message et du temps de réponse du serveur)
   - Vol d'identifiant de session par essai/erreur, écoute sur le réseau, lecture dans fichier de log, injection de commande,
   - Hameçonnage -> attaquant fournit un id de session à l'utilisateur et se connecte ensuite avec cet id

b. Authentification insuffisante
   - Pour open source, utilisation de comptes connus
   - Accès non sécurisé aux répertoires

c. Mauvais traitement des recouvrements de mot de passe
   - Procédure avec demande de renseignements personnels facilement récupérables sur réseaux sociaux

3a: Protections pour Attaque sur Authentification

a) Type brute force : Emploi d'un processus automatique pour trouver les informations protégeant un système (login, mot de passe, clé cryptographique).

Protections coté applications :

- Limitation durée de vie des sessions
- Période d'inactivité maximale à définir puis expiration de la session

Protection coté client

- Eviter la prédiction de numéro de session : Id de session doit être long
- Consulter les journaux d'activité à la recherche d'évènements inhabituels

3a: Protections pour Attaque sur Authentification

a) Type brute force : Emploi d'un processus automatique pour trouver les informations protégeant un système (login, mot de passe, clé cryptographique).

Autres Protections :

- Mots de passe forts, Nbre essais limité pour authentification
- Logger les tentatives de connexions
- Ne jamais informer de l'origine de l'erreur
- Imposer une taille et complexité minimum pour les passwords (min 8 caractères, un chiffre, lettre maj et min)
- Ne pas conserver les passwords par default
- utiliser le module mod_evasive sous apache : blocage au bout de n tentatives déverrouillage après laps de temps

3a: **Attaque sur Authentification**
Mode evasive d'apache

Mode evasive :

- Module apache , se configure dans apache.conf
- Détecte les un trop grand nombre de pages sur un site web, sur un délai de temps très court,

- S'utilise conjointement avec iptables pour blocage des IP.

```
DOSHashTableSize 3097
DOSPageCount 3
DOSSiteCount 50
DOSPageInterval 2
DOSSiteInterval 2
DOSBlockingPeriod 300
DOSEmailNotify "admin@majorxtrem.be"
DOSLogDir "/var/log/mod_evasive/"
#DOSSystemCommand "/sbin/iptables -I INPUT -s %s -j DROP"
DOSSystemCommand "/bin/echo %s >> /var/log/mod_evasive/dos_evasive.log && /bin/date >> /var/log/r
DOSWhiteList 127.0.0.1
```

l'ip sera bloquée pendant 300s si on demande 3 fois la même pages dans un intervalle de 2s.

l'ip sera bloquée pendant 300s si elle accède 50 fois au site dans un intervalle de 2s.

Log de l'action et/ou blocage de l'adresse

l'utilisateur "www-data" doit avoir les droits pour manipuler IPTables
Le dossier /var/log/mod_evasive/ doit appartenir a www-data

3b: Attaque sur Authentification

b) Authentification insuffisante pour accès intranet

- Depuis un listing de répertoire sans index.html
- Par une attaque brute force sur les noms de répertoire `/admin`, `/administrateur`, `/intranet`, `/backoffice`
- Depuis un bookmark sur un PC en libre accès
- En notant l'URL de connexion d'un admin

- Protections coté admin ou utilisateurs :
  - Configurer le serveur pour une authentification des utilisateurs (configuratio fichier `/etc/apache2/sites-availaible/default`)

  - Utilisation d'un `.htaccess` pour protéger les rep, sous répertoires et les sessions

3b: Attaque sur Authentification

Autorisation htaccess sous apache

- fichier de config des sites web

`/etc/apache2/sites-available/default`

```
<Directory /var/www/test>
        AuthType Basic
        AuthName "autorisation requise"
        # (La ligne suivante est facultative)
        AuthBasicProvider file
        AuthUserFile /var/www/admin/.htpass
        Require user admin
</Directory>
```

Rendez-vous à: http://votre.domaine.tld/test

3b: Attaque sur Authentification

.htaccess

- **AuthUserFile** définit l'emplacement du fichier contenant les logins et les mots de passe des utilisateurs autorisés à accéder à un répertoire donné.
- **AuthGroupFile** définit l'emplacement du fichier contenant les groupes d'utilisateurs autorisés à s'identifier. (optionnel)

```
<Directory /repertoire/a/proteger>

ErrorDocument 403 http://www.system-linux.eu/acces-refuse.php
AuthUserFile /repertoire/de/votre/fichier/.FichierDeMotDePasse
AuthGroupFile /dev/null
AuthName "Accès sécurisé au site CCM"
AuthType Basic

<LIMIT GET POST>
Require valid-user
</LIMIT>

</Directory>
```

3b: Attaque sur Authentification

.htpasswd

- Gère l'accès à certains fichiers et répertoire par un mot de passe.
- S'utilise avec .htaccess qui contient les passwords des utilisateurs.

```
mkdir /var/www/admin
cd /var/www/admin
htpasswd -c .htpass admin
New password:
Re-type new password:
Adding password for user admin
```

3c: **Attaque sur Authentification**

c) Mauvais recouvrement des mots de passe :

- Partage de secret : questions personnelles à la création du compte. Nécessite le stockage d'informations personnelles sur le serveur. Ca ne doit pas être une adresse, tel, secu …) qui peuvent être fournies par un pirate connaissant la victime.
- Envoi d'un nouveau MdP à l'adresse mail fournie lors de l'inscription. Attention à l'écoute !

- Protections :
  - Fournir un lien pour recréer le MdP ou si envoi d'un Mdp il doit être a usage unique.
  - Conserver les demandes de recouvrement
  - Limiter la durée du nouveau mdp ou lien à 24h