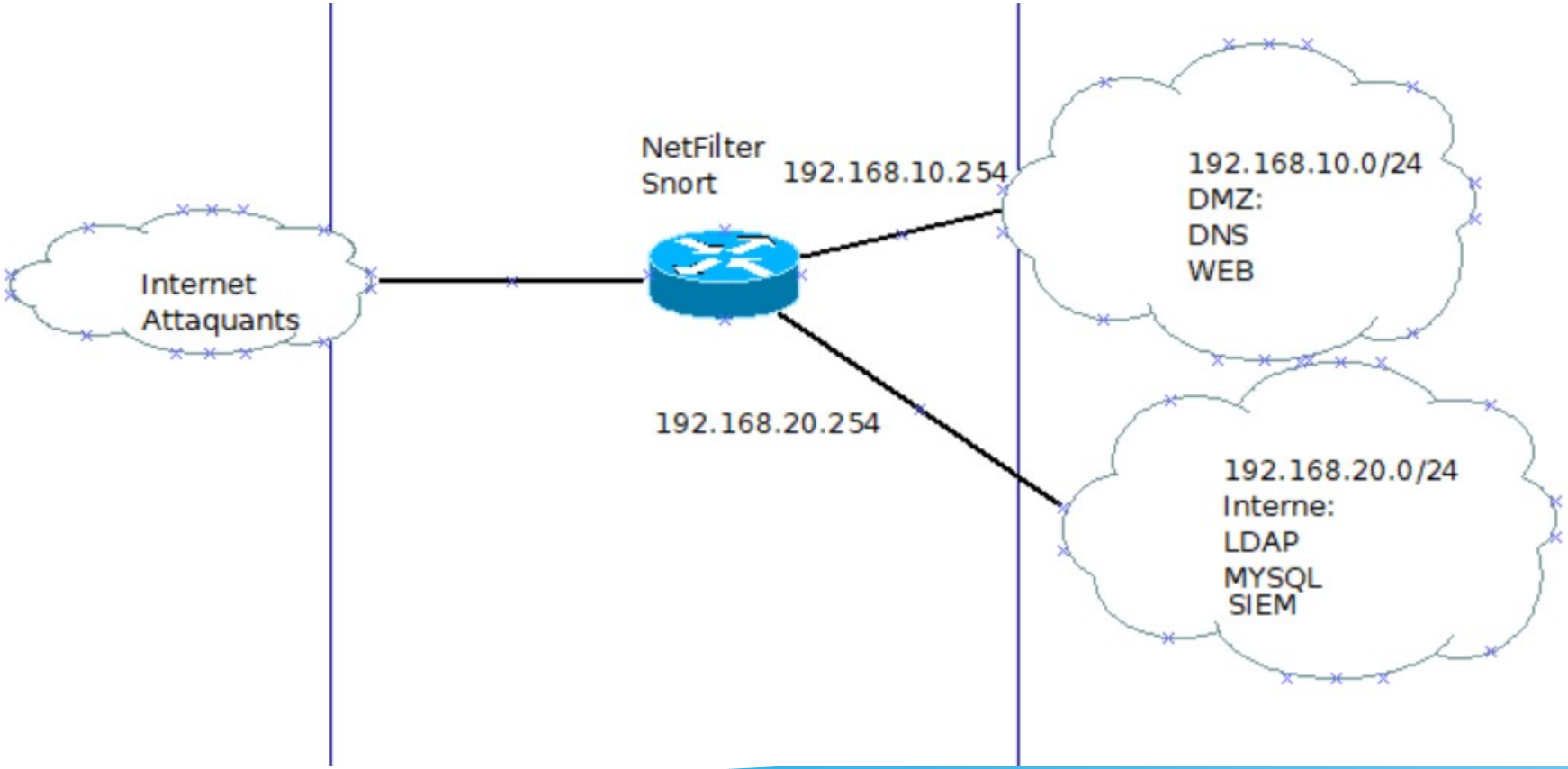
SAE Cybersécurité

Infrastructure du projet



Les étapes pour un test d’intrusion

- La reconnaissance : Collecte de renseignements à l’aide de différentes techniques et outils.

  2 moyens pour pénétrer le système :

  - depuis l’intérieur (clé USB, personne malveillante à l’intérieur,...)
  - depuis l’extérieur (Falsification d’un site Web, cheval de Troie,injections Web)
- L’énumération : Identification et évaluation des vulnérabilités.
- L’exploitation : Les vulnérabilités trouvées en amont sont exploitées → Maintien de l’accès
- L’escalade des privilèges : le pentester va chercher pénétrer de manière transversale (rebonds et exploiter d’autres failles
- Enoncer les contre-mesures

Les machines utilisées pour le projet

- Kali avec des outils d’attaque et un site Web pour récupérer les résultats des attaques
- IDS/IPS type stack ELK (Elasticsearch, Logstash, Kibana) + centralisation des logs
- Machine vulnérable à construire (LAMP) avec VirtualHosts et différents noms de domaine + authentification avec LDAP pour certaines parties du site
- Une machine cliente victime
- Serveur DNS + Serveur DHCP
- Serveur de ressources accessible depuis le serveur Web

Outils à maîtrisés

- Commandes d’accès à distance : SSH, LdapSearch, SmbClient, FTPClient, Nmap, Nikto, DirSearch, …
- Configuration d’un annuaire (Requêtes LDAP), Configuration d’un SGBDR (requêtes SQL)
- Administration d’un site dynamique avec Apache-Mysql (ou MariaDB)
- Authentification avec un annuaire LDAP
- HTML5 et CSS3, PHP + Javascript
- Burp, Hydra, scripts en Python
- Détection des intrusions avec SNORT + filtrage avec NetFilter

Les sites à visiter

- Pour les labs : https://portswigger.net
- Pour la programmation WEB : https://www.pierre-giraud.com
- https://owasp.org/
- https://www.root-me.org/
- https://tryhackme.com/

Description d’une attaque système (de l’intérieur)

- Analyse de trames pour récupérer la passerelle
- Découverte du réseau à l’aide d’outils comme NMAP (balayage des ports) pour trouver les serveurs, énumération NetBios (nbtstat), énumération SNMP (NetScanToolsPro)
- Reconnaissance LDAP, énumération SMB (infos sur les utilisateurs, partages ouverts, etc …, identification des versions), extraction d’informations (emails, mails, transferts de zones DNS,noms d’utilisateurs via SNMP), recherche de comptes oubliés, énumération SAM-R
- Visualiser un chemin d’attaque dans un environnement Active Directory (BloodHound) → Atteindre le groupe des administrateurs de domaine
- Attaque de l’homme du milieu pour détourner les flux
- Attaque sur un serveur (exploitation des failles repérées)
- Récupération d’informations pour élever les privilèges et rebondir sur une autre machine
- Extraction des données

Attaques à mettre en place sur une plateforme WEB(depuis Internet)

- Attaque par force brute sur différents protocole d'accès à distance
- Attaques par injections (avec un outil comme BURP par ex):

- Injections de commandes :données envoyées au shell via les fonctions exec, pathtru ou system

- Cross-Site Scripting (XSS) : données envoyées à la base de données sans validation via des injections dans du code HTML ou Javascript

- Injections LDAP ou Xpath (langage de requêtes XML)

- Inclusions de fichiers (utilisation de fonctions comme *include*)

- Attaques XXE (Interférence via un fichier XML)
- SSRF (Interaction avec des ressources internes via des URLs)
- Introduire un Reverse Shell pour l'attaque à distance (faux lien)

Mise en place du projet

- Mise en place de l'infrastructure
- Choix d'un scénario d'attaque (Page Web, Service SMB, cheval de Troie avec Reverse Shell, …)
- Installation des services (avec des failles)
- Installation d'éléments qui permettent l'élévation de privilèges (anciens mails, anciens comptes) et construction du chemin d'attaque
- Test de l'attaque avec différents outils
- Mise en place d'un IDS au sein d'un réseau local avec des contre-mesures ou sécurisation du serveur et du code pour une application Web

Exemples d’indices à laisser pour chaque attaque

- Laisser l’affichage des fichiers d’un répertoire s’il n’y a pas de fichier index (test avec ZAP)
- Laisser des fichiers dans l’arborescence du serveur Web permettant de faire un reverse shell
- Laisser des mails dans des répertoires avec des mots de passe
- Ancien comptes inactifs dans un fichier
- Trouver des comptes inactifs sur un annuaire LDAP
- Fichier de mots de passes dans un fichier PHP

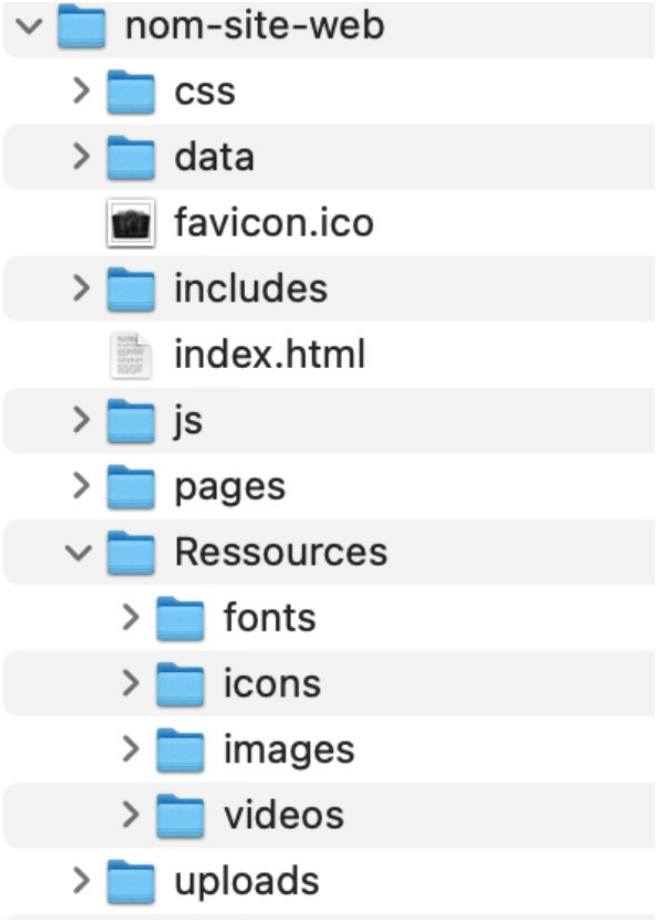
Cahier des charges du projet

- Faire un schéma de l’infrastructure
- Pas de machines pré-configurées : Installer tous les services
- Pour la partie Web, insérer du code en javascript pour la partie cliente et du code en PHP pour la partie serveur
- Tests d’intrusion
- Détection des intrusions
- Sur chaque page de challenge, un bouton pour voir le code vulnérable avec des explications sur la vulnérabilité et un bouton pour voir le code qu’il aurait fallu implémenter avec les explications

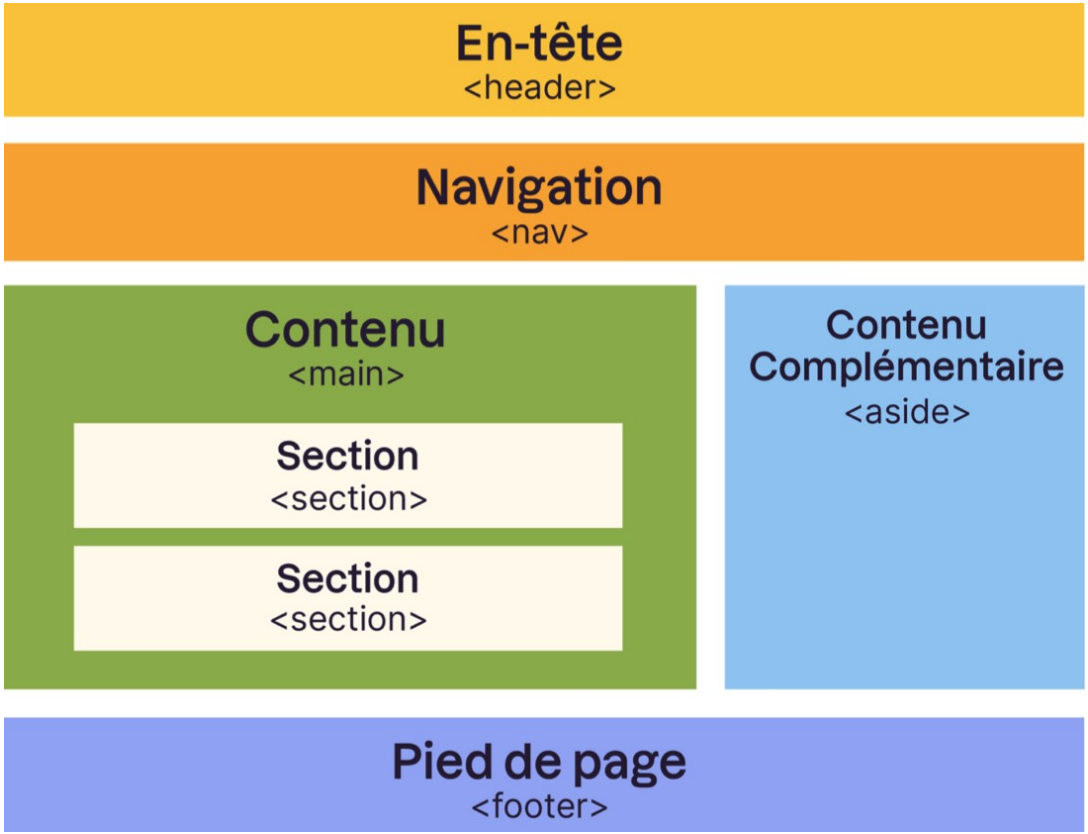
Mise en place de l’infrastructure

- Services à installer : SSH, SAMBA, FTP, APACHE, MYSQL, LDAP avec des ACLs
- Outil de détection des intrusions (SNORT, SURRICATA) + SIEM + serveur de logs

Arborescence du site



Architecture des pages Web



Organisation

- Groupes de 4 étudiants
- Partager la conception des composants du site Web
- Répartir les différentes parties du code

Travail pour chaque étudiant

- Au moins 1 script en javascript avec gestion des événements et modification du DOM
- Au moins 1 page de challenge avec du PHP
- Au moins 3 tables sous MYSQL
- Créer deux vulnérabilités
- 1 attaque depuis l’extérieur via un Reverse Shell
- Ecrire des règles avec SNORT
- Récupérations sur le SIEM + Analyse des logs
- Compte-rendu du travail personnel

Travail pour le groupe

- Choisir le contexte
- Faire le schéma de l’arborescence du site
- Choisir le contexte des vulnérabilités
- Définir plusieurs chemins d’attaque
- Définir très précisément les failles
- Définir la finalité des attaques
- Répartir les tâches et les différentes parties du code
- Planifier le travail (avec Trello par exemple)

Méthode pour la programmation PHP

- Faire un programme Test.php pour vérifier l’interpréteur
- Faire un programme Test_connexion.PHP pour vérifier Mysql
- Tester la requête sous MySQL
- Créer un programme spécifique pour la requête
- Eventuellement, éditer ou modifier la page formulaire.html correspondante
- Faire le test avec un affichage d’un résultat

Par où commencer ?

- Créer des scripts en PHP pour manipuler MySQL
- Créer des scripts en Javascript pour manipuler le DOM
- Créer des ACLs sur LDAP en fonction des utilisateurs
- Coupler Apache à OpenLDAP pour l’authentification
- Mettre en place un partage de fichiers accessible depuis Apache
- S’initier à NetFilter(IpTables)