**Institut Universitaire de Technologie,**
**Aix-Marseille Université**

**RAPPORT DE STAGE de fin de deuxième année**
**Bachelor Universitaire de Technologie**
**Spécialité Réseaux et Télécommunications**
**parcours cybersécurité**

# ÉVALUATION DES VULNÉRABILITÉS DANS LES PROJETS INDUSTRIELS

Noémie MANCINI

SNCF

Responsable entreprise : Julie DIONIGI

Responsable académique : Corinne HOUSSAIN

**2023**

# Table des matières

# 1 Introduction

Au cours de mon stage à la SNCF au sein de la Direction Cyber Sécurité, j'ai eu l'opportunité d'explorer divers aspects de la sécurité informatique. Ce rapport présente les travaux réalisés durant cette période et les résultats obtenus.

J'ai été chargée de mener à bien l'audit complet d'un système d'information industriel, depuis la phase des tests jusqu'à la rédaction du rapport. En parallèle, j'ai également assuré une veille cyber en établissant un tableau des versions logicielles et des protocoles maintenus à date.

Ce rapport est organisé en quatre axes principaux : la présentation de l'entreprise et du contexte du stage, la méthodologie de travail appliquée, ma contribution à la veille en cybersécurité, et enfin, la restitution des résultats aux industriels.

# 2 Contexte du stage

## 2.1 Présentation de l'entreprise

La SNCF, acronyme de Société Nationale des Chemins de Fer français, est un acteur majeur dans le domaine du transport ferroviaire en France et en Europe. L'entreprise a été créée en 1938 en tant qu'entreprise publique chargée de gérer le réseau ferroviaire français. Depuis sa création, elle a joué un rôle essentiel dans le développement des transports en France, facilitant la mobilité des personnes et des marchandises à travers le pays. Au fil des années, la SNCF a évolué pour devenir un groupe diversifié, comprenant plusieurs filiales spécialisées dans le transport ferroviaire de passagers, le transport de marchandises, la gestion de l'infrastructure et les services connexes.

La mission principale de la SNCF est de fournir un service de transport ferroviaire sûr, fiable et efficace à ses clients. Elle s'engage à offrir des solutions de mobilité durables, en contribuant à la réduction de l'empreinte carbone et en favorisant l'intermodalité. Elle vise également à promouvoir le développement économique des territoires en assurant la connectivité des régions, en créant des emplois et en soutenant l'innovation.

L'entreprise exerce ses activités à travers différents domaines clés. Tout d'abord, elle exploite des services de transport ferroviaire de passagers à grande vitesse, tels que le célèbre TGV ( Train Grande Vitesse), ainsi que des trains régionaux et interurbains. Ces services permettent de relier les grandes villes françaises et européennes, offrant aux voyageurs des moyens de transport pratiques et rapides. De plus, elle est également un acteur majeur dans le transport de marchandises, assurant la circulation de fret à travers le réseau ferroviaire. Cette activité contribue à désengorger les routes, à réduire les émissions de CO2 et à promouvoir le transport durable.

En outre, l'entreprise est responsable de la gestion, de l'entretien et du développement de l'infrastructure ferroviaire en France, comprenant les voies, les gares et les installations connexes. Elle s'emploie à moderniser constamment le réseau, à améliorer la sécurité et à optimiser l'expérience des voyageurs.

C'est une entreprise complexe, composée de différentes parties qui contribuent à son fonctionnement global. Voici ses principales divisions :

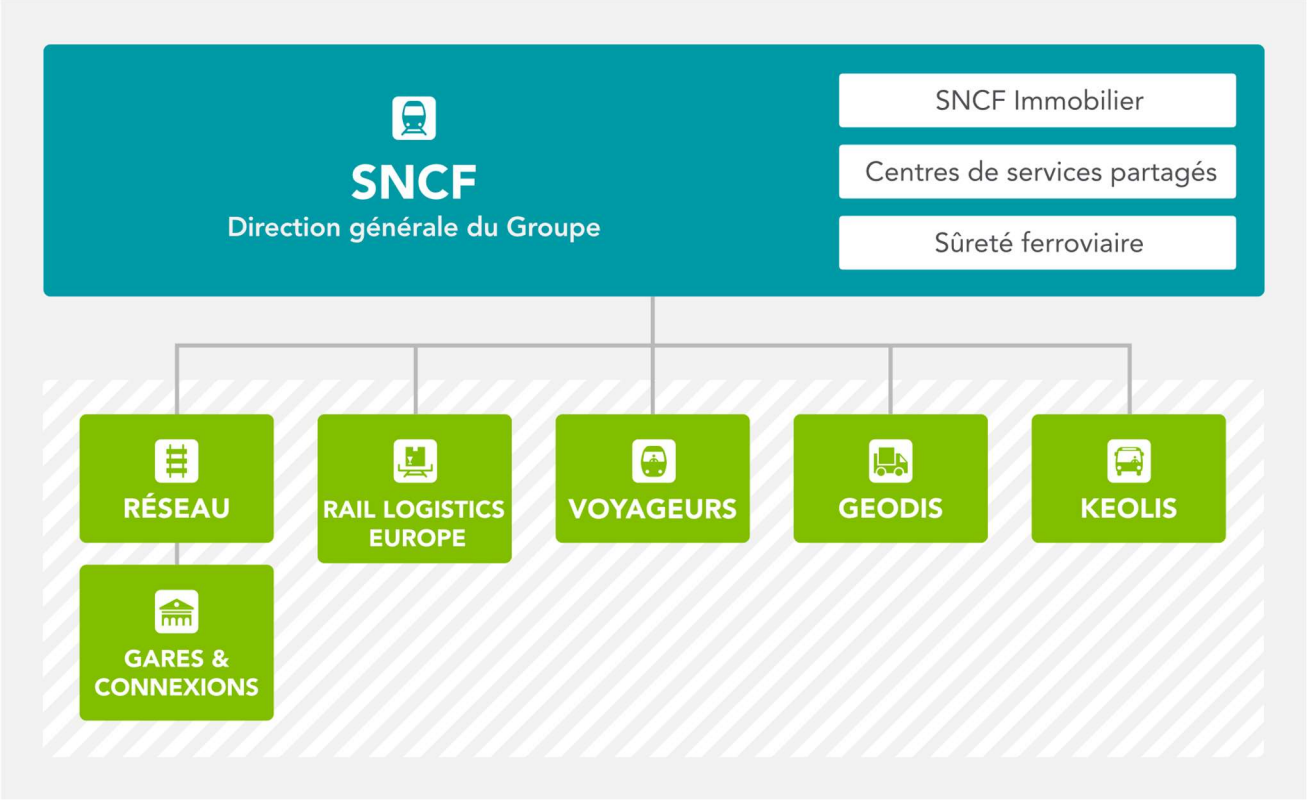


**Figure 1 - Organisation de la SNCF depuis le 1er janvier 2020**

## 2.2 Présentation du service

Pour se prémunir contre des attaques cyber qui auraient de lourdes conséquences sur ses activités, l'entreprise a créé la DCS (Direction Cybersécurité). Elle met en œuvre toute une série de mesures, depuis la veille sur les risques émergents de cybersécurité (captation de signaux faibles, veille sur le «dark web», etc.) jusqu'à la mobilisation d'équipes en charge de détecter les incidents et de circonscrire les attaques lorsqu'elles se produisent.

Leur mission consiste à positionner la Cybersécurité comme un acteur incontournable de la sécurité de l'exploitation ferroviaire du fait de la digitalisation croissante des activités des métiers. Les enjeux sont divers. Le principal étant de se défendre face aux cyberattaques, celles-ci générant 3 risques majeurs :

- L'indisponibilité totale ou partielle des systèmes d'information et/ou systèmes industriels
- La perte d'intégrité de la donnée
- La fuite ou vol de données sensibles


**Figure 2 - Les domaines d'intervention de la DCS**

Pendant mon stage, j'ai évolué au sein du département Ingénierie & Services Cyber Sécurité Industrielle, dont les objectifs principaux sont les suivants :

- Être en appui dans l'intégration de la Cybersécurité dans les projets en fournissant une expertise en matière de cybersécurité dès les phases initiales jusqu'à la mise en exploitation des systèmes. Cela inclut l'identification des risques potentiels, la définition de mesures de sécurité appropriées et l'accompagnement des équipes de projet tout au long du processus de développement puis déploiement.

- Assurer et développer l'expertise en vue de conseils et prescriptions.

- Réaliser des revues cybersécurité : réalisation de revues techniques et organisationnelles pour évaluer le niveau de cybersécurité des différents systèmes d'informations, équipements et processus.

- Être en appui dans les travaux de normalisation et de recherche : le département contribue aux travaux de normalisation et de recherche menés au niveau européen ou international.

- Ce département est focalisé sur la cybersécurité des systèmes industriels, tels que la signalisation ferroviaire (postes d'aiguillages, interfaces de gestion des circulations, …), la traction électrique (sous-stations d'alimentation), et les réseaux industriels.

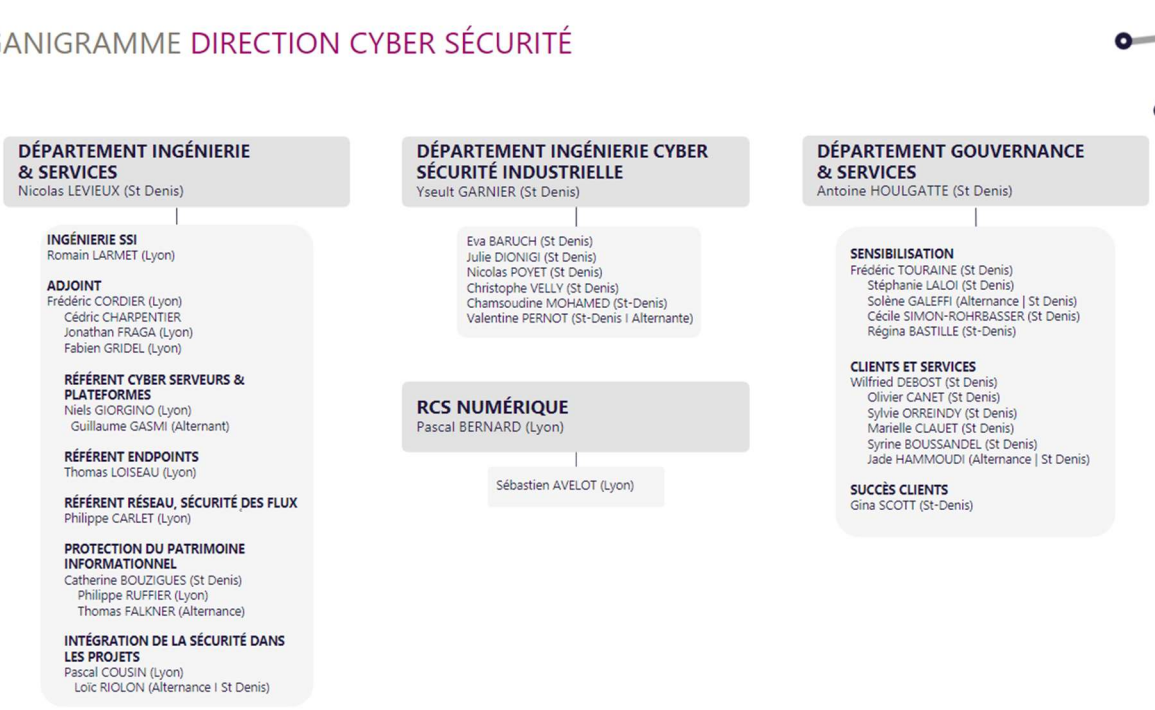


**Figure 3 – Extrait de l'organigramme de la DCS**

# 3 Méthodologie de travail

## 3.1 Accompagnement du projet et début des tests COVASEC

Pour lancer un projet, plusieurs processus d'homologation et de validation doivent être suivis. Une étape clé est la pré-qualification et l'analyse des risques, permettant de fournir les exigences de cybersécurité au projet. En fonction de sa complexité, un accompagnement adapté sera nécessaire.

La planification des tests constitue une étape importante, impliquant le remplissage du formulaire d'initialisation du **COVASEC** (Comité de Validation en Cybersécurité) ainsi que la collecte de données d'entrée. Ces éléments, préalablement fournis par l'équipe du projet, sont validés lors de notre arrivée, où des membres du projet sont présents pour nous accueillir et répondre à nos questions.

| | Simple | Standard | Complexe | Homologué | |
|---|---|---|---|---|---|
| Analyse de l'architecture | ✓ | ✓ | ✓ | ✓ | Sur la base du schéma d'architecture du système |
| Surface d'attaque | ✓ | ✓ | ✓ | ✓ | Utilisation outil « nmap », scan des ports TCP/UDP |
| Audit des vulnérabilités | ✓ | ✓ | ✓ | ✓ | Utilisation scanner de vulnérabilité, à jour de toutes les vulnérabilités connues |
| Audit de configuration | ✓ | ✓ | ✓ | ✓ | Tests d'intrusion physique et via configuration logicielle |
| Tests d'intrusion | | ✓ | ✓ | ✓ | Utilisation outil d'intrusion et brute force |
| Fuzzing | | | ✓ | ✓ | Utilisation outil fuzzing, découverte de vulnérabilités 0 Day |

**Figure 4 - Accompagnement en fonction du niveau de risque**

Une fois ces étapes préliminaires franchies, nous procédons à la réalisation des tests dans les jours ou les semaines qui suivent. Le nombre et les techniques de tests utilisés dépendent du niveau d'accompagnement déterminé lors de la préqualification **SSI** (Sécurité des Systèmes d'Information) du projet.

Ces tests que nous réalisons sont communément appelés "tests COVASEC". Ils englobent différents types de tests, chacun ayant ses propres objectifs et méthodologies.

« **Pré-COVASEC** » ► Version intermédiaire livrée par le fournisseur, pas de raccordement prévu sur réseau de production avant prochaine version :

- Echanges préalable pour évaluer la pertinence de refaire des tests sur chaque équipement,
- Déroulement de tous les tests sauf fuzzing,
- Réunion une semaine après la fin de la session de tests pour échanges avec le fournisseur et le projet et les actions à mener pour se mettre en conformité

« **COVASEC intermédiaire** » avant essais sur réseau de production ► Version intermédiaire livrée par le fournisseur, essais à prévoir sur réseau de production hors exploitation

- Déroulement de tous les tests sauf fuzzing,
- Aucune modification du système n'est envisageable en cours de COVASEC,
- Echange avec le fournisseur suite à réception du rapport, participation aux COTECH/CHOM pour décisions RCS

« **COVASEC Final** » avant mise en exploitation du système ► Version finale livrée par le fournisseur pour mise en exploitation sur réseau de production

- Idem ci-dessus + fuzzing

Pour mener à bien nos tests COVASEC, notre équipe utilise des ordinateurs de marque « Getac ». Leur principal avantage réside dans leur performance, leur robustesse et leur portabilité, ce qui nous permet de les transporter facilement et de les utiliser dans divers environnements.


**Figure 5 - Getac ordinateur utilisé pour les tests**

Voici les prérequis nécessaires pour chacun de nos tests, qui constituent en partie nos données d'entrée :

- Le formulaire d'initialisation comprenant les détails des équipements et des protocoles utilisés sur chacun d'entre eux.
- Pour chaque interface IP de chaque équipement : son adresse IP, son masque de sous-réseau et sa passerelle.
- Le fichier requis pour l'audit de configuration, notamment les identifiants de connexion (login/mot de passe) de chaque équipement.
- Pour chaque équipement à tester :
  - 2 adresses IP dans le sous-réseau de l'équipement pour connecter nos équipements de test.
  - Un port RJ45 actif.

- La désactivation des pare-feux (périmétriques et internes).
- Si possible avant les tests, prévoir des VLAN communs au maximum d'équipement à tester afin de réduire le temps de tests (cela permet de paralléliser les scans).
- Les équipements à tester doivent être paramétrés comme la cible en production et les services détectés lors des scans seront comparés à ce qui a été déclaré dans les formulaires.
- Une personne qui a la main sur les équipements doit être présente le premier jour de nos tests et lors de l'audit de configuration.
- La phase de pentest réalisée par l'équipe intervient à la fin de la phase de validation fonctionnelle en laboratoire et en amont du raccordement des équipements sur le réseau de production. La validation par le RCS du rapport de tests permet d'autoriser le projet à se raccorder aux réseaux télécom de production et de poursuivre son déploiement.
- Il est courant que le projet doive procéder à des modifications de ses systèmes suite à l'audit cybersécurité avant toute mise en service sur les réseaux de production. Un nouvel audit est alors réalisé pour vérifier que le système est conforme.

## 3.2 Scans réseau avec Nmap

Dans le cadre du COVASEC, nous avons utilisé l'outil Nmap afin de réaliser des scans réseau. Il s'agit d'un puissant outil de reconnaissance de réseau permettant d'identifier les hôtes actifs, les ports ouverts et les services en cours d'exécution. Il utilise différentes techniques de scanning pour interroger les systèmes cibles et recueillir des informations sur leur configuration.

### 3.2.1 Techniques de scan

Lors des scans réseau avec Nmap, il est important de prendre en compte deux protocoles principaux : **TCP** (Transmission Control Protocol) et **UDP** (User Datagram Protocol). Ces protocoles sont utilisés pour acheminer les données à travers les réseaux, mais ils diffèrent dans leur mode de fonctionnement.

Le scan TCP consiste à envoyer des paquets de demande de connexion TCP à chaque port du système cible, puis à analyser les réponses pour déterminer si le port est ouvert, fermé ou filtré. Cela permet d'identifier les services actifs et les vulnérabilités potentielles associées aux ports ouverts.

Le scan UDP, quant à lui, envoie des paquets UDP vers des ports spécifiques et attend des réponses. Il est utilisé pour détecter les services fonctionnant sur le protocole UDP, tels que les services de streaming ou de voix sur IP. Cependant, les scans UDP peuvent être plus complexes en raison de la nature sans connexion de ce protocole et des éventuelles réponses manquantes.

Le *3-way handshake* TCP est un processus de négociation de connexion utilisé lors de l'établissement d'une communication TCP entre un client et un serveur. Il se déroule en trois étapes :

- Le client envoie un paquet **SYN** (synchronize) au serveur pour demander l'établissement d'une connexion.
- Le serveur répond en envoyant un paquet **SYN-ACK** (synchronize-acknowledge) pour confirmer la demande de connexion et fournir son propre numéro de séquence.
- Le client envoie un dernier paquet **ACK** (acknowledge) pour confirmer la réception du SYN-ACK et établir la connexion. À ce stade, les deux parties sont prêtes à échanger des données.

Le 3-way handshake TCP garantit une synchronisation appropriée et fiable entre le client et le serveur, assurant ainsi une connexion stable et sécurisée ce qui est l'inverse de l'UDP.

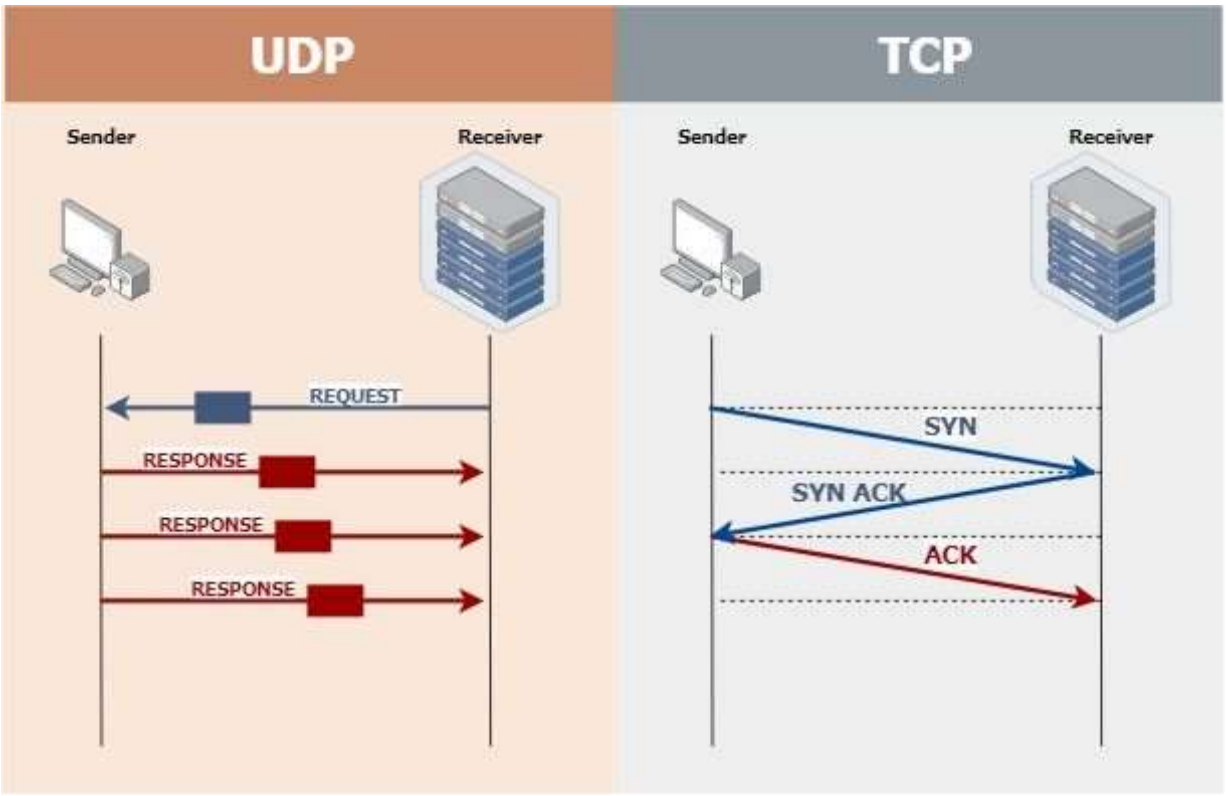


**Figure 6 - Protocole TCP et UDP**

Le fait que nous n'effectuons pas ces scans TCP et UDP en même temps est principalement liée à la complexité et à la performance. Lorsqu'on combine les deux types de scans, il peut être difficile de traiter les réponses et de distinguer clairement les résultats pour chaque protocole. De plus, certains ports peuvent réagir différemment aux scans TCP et UDP, ce qui nécessite une analyse distincte pour chaque protocole.

Outre les scans TCP et UDP, il existe d'autres types de scans possibles avec Nmap. Par exemple, le scan SYN, également connu sous le nom de "half-open scan", exploite la propriété du protocole TCP qui permet d'établir une connexion partielle sans effectuer « la poignée de main complète ». Ce type de scan peut être utile pour identifier les hôtes ou les ports actifs tout en minimisant le trafic réseau. Il s'agit de l'argument le plus utilisé.

Il existe également des scans « Xmas » « Fin » et « Null » qui exploitent des comportements particuliers des protocoles TCP pour tenter de détecter des hôtes ou des ports ouverts. Ces scans s'appuient sur l'envoi de paquets avec des drapeaux TCP spécifiques pour observer les réponses et identifier les états des ports.

Chaque type de scan présente des avantages et des inconvénients en termes de performances, de précision et de discrétion. Le choix du type de scan dépendra des objectifs spécifiques de l'évaluation de la sécurité et des contraintes liées au contexte de l'entreprise.

J'ai proposé à l'équipe d'utiliser des techniques de scans réseau avancées, notamment les scans XMAS NULL et FIN, en complément des scans traditionnels tels que TCP et UDP. J'ai constaté que ces combinaisons moins courantes pouvaient révéler des vulnérabilités supplémentaires qui étaient parfois négligées par les scans principaux. En outre, j'ai également exploré des moyens d'optimiser les scans en ajoutant des options telles que l'option *-n*, qui permet de désactiver la résolution **DNS** (Domain Name System), afin d'accélérer le processus de scan et d'obtenir des résultats plus rapidement. Ces efforts visant à améliorer l'efficacité et la précision des scans ont permis de renforcer notre capacité à détecter et à évaluer les vulnérabilités des systèmes cibles.

### 3.2.2 Résultats d'analyse

Voici une fois les scans effectués les résultats mis en forme sous forme de tableau que nous exploitons. Ci-dessous, un exemple de résultat suite au scan de ports :

| Equipement | Proto | Port | Etat | Service | Raison | Version |
|---|---|---|---|---|---|---|
| 10.3.3.3 | TCP | 22 | open | ssh | syn-ack | OpenSSH 7.6p1 Ubuntu 4ubuntu0.6 Ubuntu Linux; protocol 2.0 |
| 10.3.3.3 | TCP | 80 | open | http | syn-ack | |
| 10.3.3.3 | TCP | 81 | open | http | syn-ack | nginx |
| 10.3.3.3 | TCP | 4567 | open | http | syn-ack | nginx |
| 10.3.3.3 | TCP | 8060 | open | http | syn-ack | nginx 1.16.1 |
| 10.3.3.3 | UDP | 161 | open | snmp | udp-response | SNMPv1 server public |

Ici, l'analyse des résultats des scans Nmap a révélé les informations suivantes :

- Nginx 1.16.1 : Il s'agit de la version du serveur web Nginx installée sur le système cible. Bien que la version 1.16.1 soit relativement récente, il est important de noter qu'il existe des versions plus récentes disponibles qui peuvent inclure des correctifs de sécurité importants. Par conséquent, il est recommandé de vérifier si des mises à jour sont disponibles pour garantir la sécurité du serveur web.

- SNMPv1 server (public) : Le résultat indique que le service SNMPv1 est actif sur le système cible et utilise la communauté "public". Le protocole SNMPv1 est une version ancienne et non sécurisée, qui utilise des chaînes de communauté non chiffrées pour l'authentification. L'utilisation de la communauté "public" par défaut peut permettre à des attaquants de récupérer des informations sensibles ou de modifier la configuration du système. Il est recommandé de désactiver ce service ou de passer en SNMPv3. En dernier recours, il est demandé de sécuriser correctement le service SNMPv1 en utilisant des chaînes de communauté complexes et en limitant les accès.

- OpenSSH 7.6p1 Ubuntu 4ubuntu0.6 Ubuntu Linux; protocol 2.0 : Cela indique que le système cible utilise OpenSSH version 7.6p1 sur une distribution Ubuntu Linux. OpenSSH est un protocole de communication sécurisé utilisé pour les connexions à distance. Bien que la version 7.6p1 soit relativement récente, il est essentiel de maintenir OpenSSH à jour en appliquant les derniers correctifs de sécurité. Des vulnérabilités connues peuvent être exploitées si le système n'est pas à jour.

À partir de ces informations, certaines vulnérabilités potentielles peuvent être envisagées notamment au travers des vulnérabilités connues pour la version spécifique de Nginx (1.16.1) et d'OpenSSH mais également avec l'utilisation du SNMPv1 avec la communauté "public".

## 3.3 Tests de vulnérabilité avec Besecure

L'outil BeSecure est un scanner de vulnérabilité utilisé dans le domaine de la cybersécurité. Il permet d'identifier les faiblesses et les vulnérabilités potentielles dans les systèmes informatiques, les applications et les réseaux. En utilisant diverses techniques d'analyse, BeSecure effectue des scans exhaustifs pour détecter les points faibles et les configurations non sécurisées. Le logiciel se base sur la liste des vulnérabilités connues, notamment les éléments fournis par le **CERT-FR (**Computer Emergency Response Team) et les évaluations **CVSS** (Common Vulnerability Scoring System). Il fournit ensuite des rapports détaillés sur les vulnérabilités découvertes, ce qui permet de prendre des

mesures correctives pour renforcer la posture de sécurité de leurs systèmes. Les mises à jour logicielles régulières sont indispensables afin de tenir à jour la base de données des vulnérabilités et garantir un scan des plus pertinents.

L'objectif principal du scanner de vulnérabilité Besecure est de localiser et d'identifier les vulnérabilités connues qui pourraient être exploitées par des attaquants. Il fonctionne en analysant en profondeur les systèmes cibles à la recherche de failles connues, telles que des vulnérabilités logicielles, des configurations incorrectes, des services obsolètes ou des mises à jour manquantes.

### 3.3.1 Analyse des résultats obtenus

Grâce à l'utilisation de l'outil Besecure, nous avons pu générer un rapport détaillé sur les vulnérabilités identifiées lors des analyses. Ce rapport nous fournit des informations précieuses sur les failles de sécurité détectées, ainsi que des recommandations sur les mesures à prendre pour les résoudre. De plus, Besecure nous offre la possibilité d'exporter ces données au format Nessus, ce qui sera particulièrement utile pour la prochaine étape de notre COVASEC.

Voici un exemple de tableau de résultats une fois mis en forme, que nous conservons pour référence et pour assurer un suivi des vulnérabilités identifiées :

| Equipement | Port | Criticité | Vulnérabilité |
|---|---|---|---|
| 10.3.3.3 | http (81/tcp) | Critique | Fingerprinting GitLab CE/EE ExifTool Unauthenticated RCE |
| 10.3.3.3 | http (81/tcp) | Critique | GitLab CE/EE image RCE |
| 10.3.3.3 | http (81/tcp) | Important | Web Application Cookies Lack Secure Flag |
| 10.3.3.3 | ssh (22/tcp) | Important | SSH Supports Weak Algorithms |
| 10.3.3.3 | http (81/tcp) | Important | X-Forwarded-Host Header Injection |
| 10.3.3.3 | http (81/tcp) | Important | Rails CRLF XSS |
| 10.3.3.3 | http (4567/tcp) | Faible | Remote Host Replies to SYN+FIN |
| 10.3.3.3 | https (443/tcp) | Faible | SSL Verification Test |
| 10.3.3.3 | http (81/tcp) | Faible | Identify Unknown Services via GET Requests |
| 10.3.3.3 | http (80/tcp) | Faible | HTTP Packet Inspection |

Ce tableau nous permet de visualiser clairement les vulnérabilités identifiées, leur port/service et criticité respectifs.

## 3.4 Tests de compromission avec Metasploit

### 3.4.1 Rôle de Metasploit

L'outil Metasploit Pro est une plateforme complète de tests d'exploitation et de gestion des vulnérabilités, développée par Rapid7. Il s'agit d'une version professionnelle et avancée de Metasploit proposant une interface graphique.

Metasploit Pro offre une multitude de fonctionnalités et de capacités pour réaliser des tests d'exploitation sur des systèmes et des applications vulnérables. Il permet aux professionnels de la sécurité de simuler des attaques réelles dans le but d'évaluer la résistance d'un système ou d'un réseau aux cyberattaques.

L'interface utilisateur de Metasploit Pro facilite la configuration des tests d'exploitation. Il propose également une vaste bibliothèque d'exploits, de payloads et de modules, ce qui permet aux utilisateurs de personnaliser leurs attaques en fonction des vulnérabilités identifiées.

Parmi les fonctionnalités clés de Metasploit Pro, on trouve la possibilité de réaliser des attaques automatisées, des tests d'intrusion en plusieurs étapes et des simulations d'attaques ciblées. Il permet également de générer des rapports détaillés sur les vulnérabilités identifiées, les exploits réussis et les mesures de sécurité recommandées.

### 3.4.2 Présentation des étapes suivies

Voici la méthodologie de travail que j'ai suivie :

La première étape réalisée est un scan rapide. Cette étape consiste à scanner les réseaux et à détecter rapidement les hôtes actifs ainsi que les services ouverts sur ces hôtes. Cela nous permet d'avoir une vue d'ensemble des cibles potentielles.

Après avoir effectué le scan rapide, j'ai intégré les résultats de mes analyses précédentes dans Metasploit Pro : j'ai procédé étape par étape en injectant successivement les résultats du scan TCP de Nmap, puis du scan UDP, et enfin les résultats de Besecure au format Nessus.

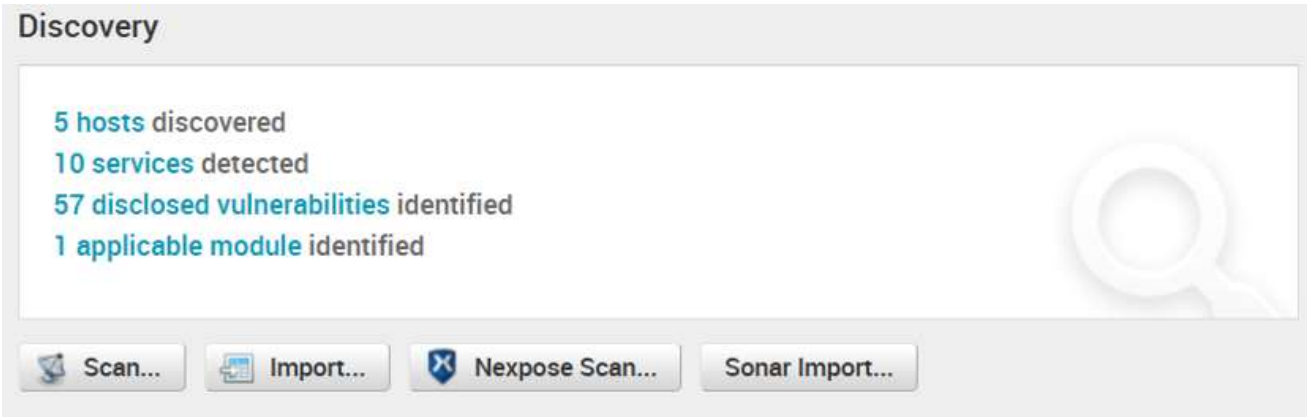


**Figure 7 - Interface récapitulative du projet**

Cette approche progressive permet d'avoir une vision globale des premières vulnérabilités présentes dans l'environnement testé.

Si un ou plusieurs services web sont détectés lors du scan rapide ou lors des autres scans, je lance alors un scan web plus approfondi. Cette étape vise spécifiquement à évaluer la sécurité des applications web en effectuant des tests d'intrusion ciblés sur ces services.

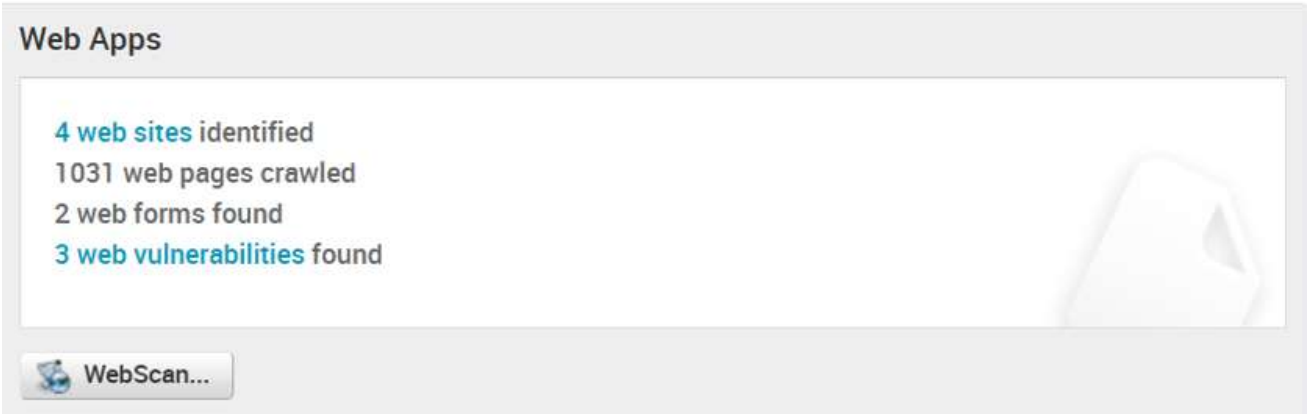


**Figure 8 – Interface de scan web après WebScan**

Cette fonctionnalité nous permet en même temps de signaler si SSL n'est pas activé et de signaler si les chiffrements SSL faibles sont autorisés.

Puis nous utilisons des techniques automatisées pour effectuer un bruteforce sur les mots de passe, en activant les mutations pour augmenter les chances de réussite. De plus, nous utilisons une liste personnalisée de mots de passe qui inclut des éléments tels que les noms de l'entreprise ou du projet, afin de cibler les éventuelles faiblesses liées à des mots de passe prédictibles.

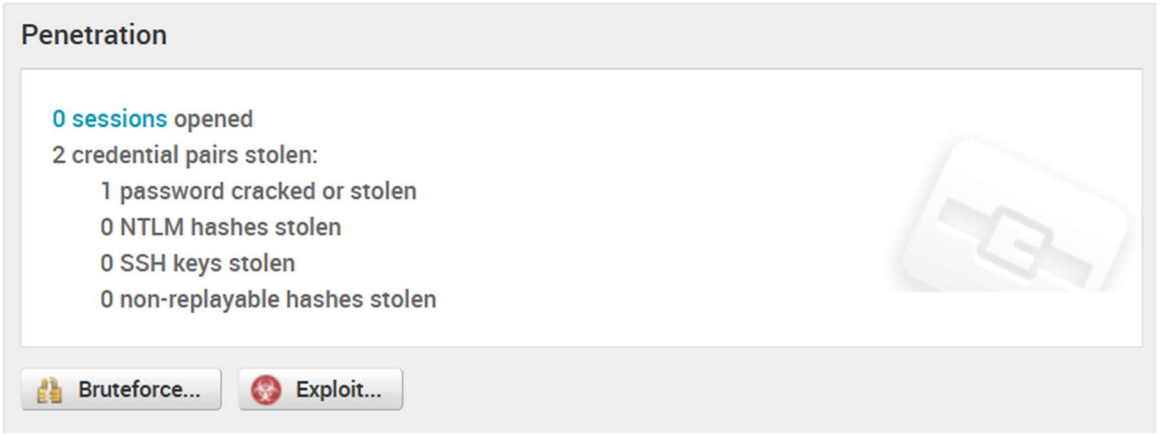


**Figure 9 – Succès de la tentative de bruteforce**

Enfin, je passe à l'étape cruciale du test d'intrusion. Cette étape permet de simuler les techniques utilisées par les hackers afin de vérifier la résistance du système et de déterminer si des failles de sécurité connues sont présentes.

### 3.4.3 Test d'intrusion

Lors de la réalisation du test d'intrusion avec Metasploit Pro, j'ai rapidement compris que ce processus ne pouvait être entrepris sans une réflexion approfondie. En effet, les paramètres entrés pour effectuer le test sont d'une importance capitale, car ils déterminent la portée et l'efficacité de celui-ci.

L'un des éléments cruciaux à prendre en compte est le choix du type de payload. Il est essentiel de sélectionner le payload approprié en fonction de la cible et des vulnérabilités potentielles identifiées. Un choix judicieux du payload peut permettre une exploitation réussie des failles de sécurité.

Le choix du payload à utiliser lors d'un test d'intrusion dépend de plusieurs facteurs, tels que le système d'exploitation de la cible, les restrictions de sécurité en place et les objectifs spécifiques du test. Très souvent, lors de nos COVASEC, nous sommes confrontés à la nécessité de tester simultanément plusieurs équipements présentant différentes caractéristiques et configurations. Cela implique de réaliser les mêmes tests d'intrusion sur ces équipements simultanément et donc avec le même choix de payload.

Parmi les options de payload couramment utilisées, nous retrouvons Meterpreter, Meterpreter 64 bits, PowerShell et Shell.

- **Meterpreter** est un choix populaire en raison de sa polyvalence et de ses fonctionnalités avancées. Il offre une interaction étendue avec le système cible, permettant de prendre le contrôle complet de la machine compromise.

- **Meterpreter 64 bits** est spécifiquement conçu pour les systèmes d'exploitation 64 bits. Il exploite les avantages de l'architecture 64 bits et offre une meilleure compatibilité avec les environnements modernes. Si la cible dispose d'un système d'exploitation 64 bits, l'utilisation de Meterpreter 64 bits peut augmenter les chances de succès de l'exploitation des vulnérabilités.

- **PowerShell** est un langage de script puissant intégré aux systèmes d'exploitation Windows. Il permet d'automatiser des tâches, d'interagir avec des services et de manipuler des données. L'utilisation de PowerShell comme payload peut être particulièrement utile dans les environnements Windows, car il exploite les fonctionnalités natives du système d'exploitation.

- Enfin, **Shell** fait référence à une interface de ligne de commande basique qui permet d'exécuter des commandes système sur la cible. Bien qu'il puisse sembler moins sophistiqué que les autres options, il peut être utilisé efficacement dans certaines situations, notamment lorsque des restrictions de sécurité limitent l'utilisation de payloads plus avancés.

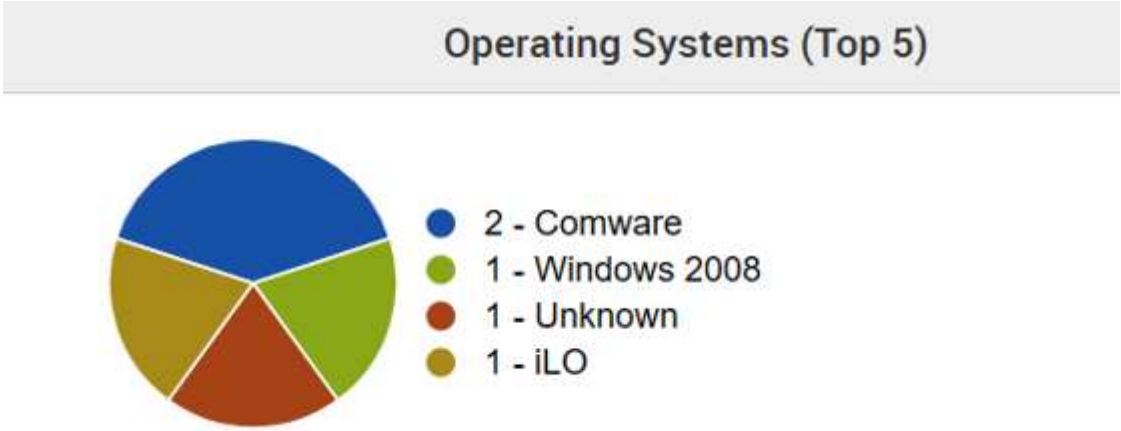


**Figure 10 - Exemple d'OS détectés simultanément lors d'un scan**

De plus, la sélection des ports d'écoute est un autre aspect important. Il faut tenir compte des services spécifiques qui sont utilisés par la cible et s'assurer que les ports appropriés sont inclus dans le test. Cela permet de maximiser les chances de détecter et d'exploiter des vulnérabilités spécifiques.

Des fonctionnalités d'évasion antivirus et d'**IPS** (Système de Prévention des Intrusions) peuvent également être nécessaires dans certains cas. L'« évasion », permet de contourner les mécanismes de défense mis en place par la cible, afin de tester réellement sa résistance aux attaques. Cependant, la plupart du temps, nous opérons en mode "whitebox", ce qui signifie que les pare-feux sont désactivés et que nous avons un accès privilégié aux systèmes que nous évaluons.

Enfin, le choix du nombre de d'exploit concurrents et du timeout en minutes doit être fait avec prudence ! Un nombre trop élevé de d'exploit concurrent peut surcharger la cible et perturber son fonctionnement normal, tandis qu'un nombre trop faible peut réduire l'efficacité du test. De même, le timeout doit être défini de manière adéquate pour permettre une exploitation suffisante des vulnérabilités identifiées.

Pendant ce stage, j'ai fait face à une situation intéressante où j'ai constaté qu'une même vulnérabilité, sur des équipements ayant la même configuration, pouvait conduire à des résultats différents. En effet, sur l'un de ces équipements, j'ai réussi à compromettre la sécurité, tandis que sur un autre, aucune vulnérabilité n'a été détectée.

Pour résoudre cette problématique, j'ai dû effectuer des ajustements manuels sur les paramètres afin de trouver la combinaison adéquate. En identifiant les différences et en adaptant les paramètres en conséquence, j'ai finalement réussi à compromettre le deuxième équipement de la même façon.

### 3.4.4 Exploitation de la compromission

Sur les systèmes que j'ai pu tester, il est rare qu'un test de compromission réussisse et qu'une session s'ouvre. Cependant, cela s'est produit dans un cas particulier. J'ai pu ouvrir une session sur un équipement en exploitant une vulnérabilité connue, également appelée **CVE** (Common Vulnerabilities and Exposures).

Cette CVE nous a permis d'obtenir une session administrative sur l'équipement réseau sans nécessiter d'authentification préalable. Suite à cette réussite, j'ai proposé à l'équipe d'aller plus loin dans l'exploitation de cette compromission afin de comprendre les possibilités et les risques associés.

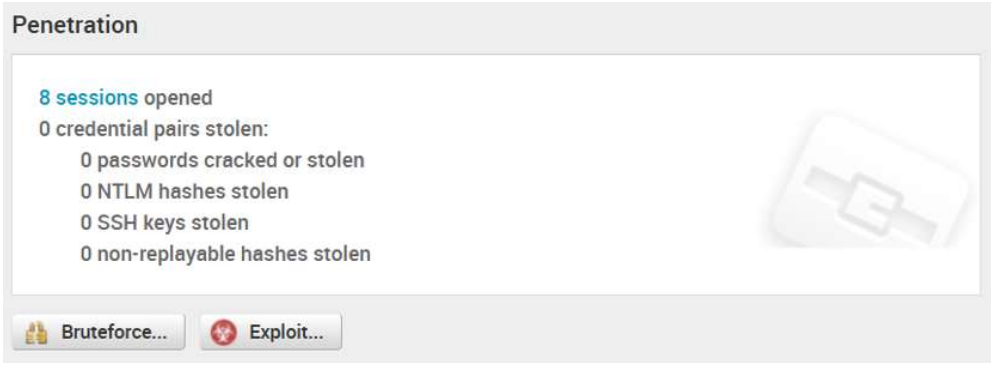


**Figure 11 - Interface test d'intrusion lors de la réussite d'une compromission**

En explorant davantage cet équipement, j'ai pu récupérer de nombreuses informations sur sa configuration. J'ai également réussi à créer un compte administrateur avec un mot de passe de mon choix. Étant donné que l'équipement disposait d'un portail d'authentification sur le port 80, j'ai pu me connecter à cette interface d'administration en utilisant mon compte administrateur nouvellement créé.

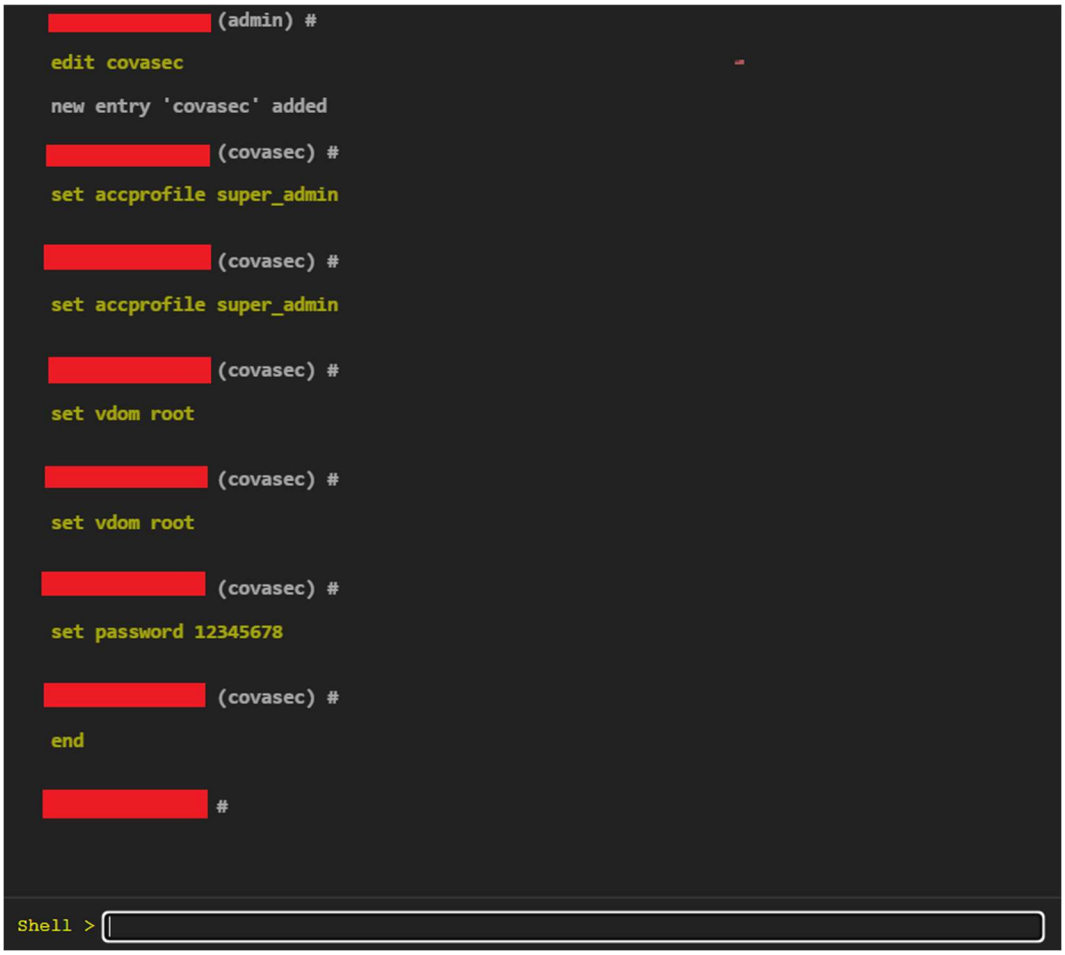


**Figure 12 - Session ouverte sur metasploit, création d'un nouvel administrateur**

À partir de ce point d'accès, une personne mal intentionnée peut exercer un contrôle total sur l'équipement lui-même, ainsi que sur tous les équipements qui en dépendent.

## 3.5 Tests d'application web avec AppSpider

### 3.5.1 L'outil AppSpider

AppSpider est un outil de test de sécurité des applications web conçu pour identifier les vulnérabilités et les faiblesses de sécurité potentielles. Il permet d'évaluer la robustesse des applications web et d'identifier les failles de sécurité avant qu'elles ne soient exploitées par des attaquants.

L'outil AppSpider fonctionne en explorant de manière exhaustive les applications web et en analysant leur comportement pour détecter les vulnérabilités courantes en se référant aux base de données des vulnérabilités telle que celle du CERT FR . Il utilise des techniques d'analyse dynamique et d'inspection de code pour identifier les problèmes de sécurité tels que les injections **SQL** (Structured Query Language), les failles **XSS** (Cross-Site Scripting), les vulnérabilités de contrôle d'accès, les erreurs de configuration, les fuites d'informations sensibles et bien d'autres encore.

Voici quelques fonctionnalités clés de l'outil AppSpider que nous utilisons :

- **Exploration automatique des applications** : AppSpider analyse automatiquement les applications web pour découvrir les pages, les formulaires, les paramètres et les fonctionnalités disponibles. Il suit les liens internes et explore l'application de manière exhaustive, en identifiant les points d'entrée potentiels pour les attaques.

- **Identification des vulnérabilité**s : Une fois que l'exploration est terminée, AppSpider effectue une série de tests pour identifier les vulnérabilités de sécurité. Il envoie des requêtes malveillantes, des injections de code et des tentatives d'exploitation pour détecter les failles et les faiblesses potentielles.

- **Analyse de la sécurité** : AppSpider évalue la sécurité de l'application en identifiant les vulnérabilités spécifiques, leur gravité et les recommandations pour les corriger. Il fournit des informations détaillées sur chaque vulnérabilité détectée, y compris des exemples d'exploitation et des conseils pour les remédiations.

- **Génération de rapports** : L'outil AppSpider génère des rapports complets qui répertorient toutes les vulnérabilités détectées, leur niveau de gravité et les mesures correctives recommandées. Ces rapports facilitent la communication avec les développeurs et les responsables de l'application, permettant ainsi de prendre les mesures appropriées pour renforcer la sécurité.

### 3.5.2 Rapport AppSpider

AppSpider Pro propose en réalité différents types de rapports que l'on peut utiliser en fonction de nos besoins :

- Structure du site, qui affiche tous les liens explorés et les vulnérabilités découvertes.
- Informations sur le site, qui fournit des métriques sur les différents objets vulnérables découverts dans votre application.
- Résultats, qui détaille les découvertes de vulnérabilités par rôle, tels que Développeur d'application ou Administrateur de serveur.

On peut utiliser les différents rapports pour accéder à l'état actuel de la sécurité de l'environnement et partager les mesures de correction recommandées avec les parties prenantes. Ensuite, selon le rôle, nous pouvons commencer à approfondir les vulnérabilités individuelles ou les catégories de vulnérabilités. L'onglet *Résultats* catégorise spécifiquement les vulnérabilités dans les domaines suivants :

- Développeur d'Application
- Administrateur de Base de Données
- Administrateur de Serveur
- Confidentialité
- Réflexion
- Bonnes Pratiques


**Figure 13 - Vulnérabilités par rôle**

Chaque vulnérabilité affiche une description, une référence et une recommandation pour la correction. On peut également identifier la requête utilisée et la réponse reçue, y compris les réponses HTML.

L'onglet *Résultats* nous permet de décomposer les résultats par rôle, par gravité ou par attaques actives et passives. Cet onglet utilise des couleurs pour fournir des informations de haut niveau : Le gris indique que le lien ou le dossier n'a pas été scanné. Le bleu indique des résultats informatifs. Le vert indique que la découverte est sûre. Le jaune indique des résultats à faible risque. L'orange indique des résultats à risque moyen. Le rouge indique des résultats à haut risque.

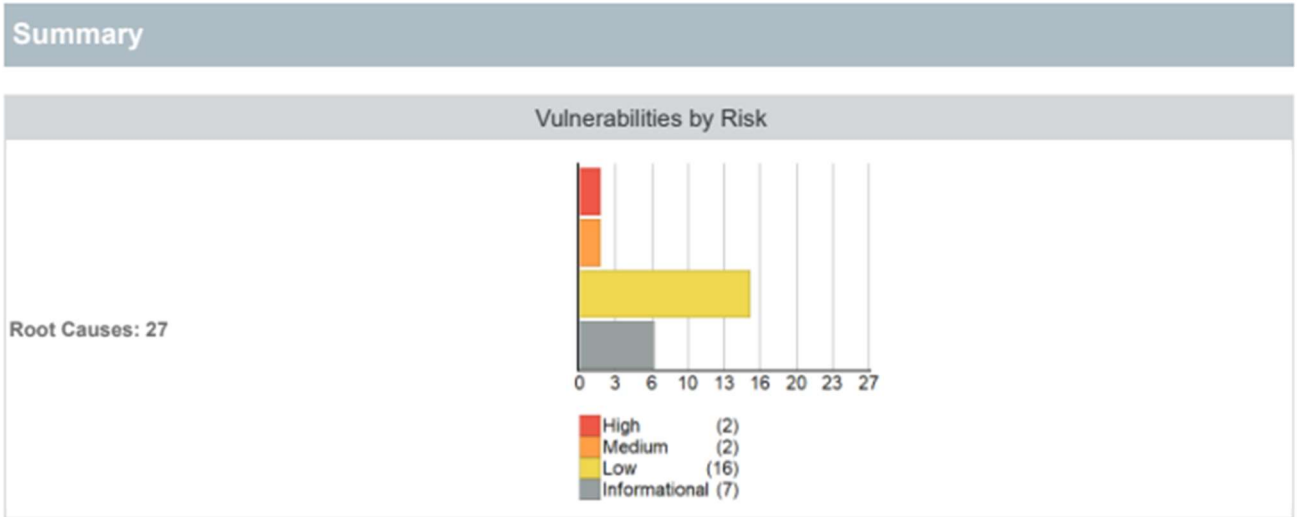


**Figure 14 - Vulnérabilités par risques**

Chaque vulnérabilité contient les informations suivantes :

- Type d'attaque
- Description de l'attaque : Description de la variante de l'attaque tentée.
- Valeur d'origine : La valeur non malveillante d'origine du paramètre, utilisée pour observer le comportement normal de l'application.
- Paramètre
- Valeur d'attaque : Une valeur spécialement conçue censée provoquer un comportement anormal de l'application.
- Vulnérabilité : la chaîne spécifique utilisée dans l'attaque.
- Trafic de crawling - Un instantané de la requête HTTP envoyée avec la valeur d'attaque dans le paramètre, et la réponse erronée qui en résulte, ainsi que de la requête HTTP envoyée à l'application et de la réponse reçue dans le cas normal.

| Vulnerability Type | Root Causes | Variances |
|---|---|---|
| Anonymous Access | 4 | 8 |
| Browser Cache directive (leaking sensitive information) | 2 | 2 |
| Content Type Charset Check | 13 | 13 |
| Credentials sent with GET method | 2 | 4 |
| JavaScript Memory Leaks | 1 | 1 |
| Predictable Resource Location | 3 | 3 |
| SSL Certificate Domain Mismatch | 2 | 2 |
| Total: | 27 | 33 |

**Figure 15 – Exemple type de vulnérabilités relevées**

L'outil laisse également le libre choix de modifier la gravité et le statut de la vulnérabilité. Par exemple, si nous estimons que le secteur est sujet à des attaques en raison d'une certaine vulnérabilité, nous pouvons modifier une vulnérabilité à faible risque pour la considérer comme un risque élevé.

## 3.6 Test de configuration et conformité

Dans le cadre de l'évaluation de la sécurité des projets en production, l'un des aspects essentiels est la réalisation de tests de configuration des équipements. Ces tests visent à vérifier si les équipements répondent aux exigences de sécurité définies à la suite de l'analyse de risques. Leur objectif est d'identifier les éventuelles faiblesses de configuration qui pourraient être exploitées par des attaquants.

Lors des tests de configuration, plusieurs aspects sont évalués. En prérequis, les équipes demandent aux correspondants projet de réactiver les pare-feux conformément à la configuration cible terrain. Tout d'abord, on vérifie si le pare-feu est activé et correctement configuré pour filtrer le trafic entrant et sortant. Un pare-feu bien configuré joue un rôle crucial dans la défense du réseau contre les attaques externes.

Ensuite, on contrôle l'utilisation sécurisée des ports USB. Les ports USB peuvent représenter une vulnérabilité potentielle si leur utilisation n'est pas restreinte ou si les politiques de sécurité associées ne sont pas en place. Il est donc important de s'assurer que les ports USB sont correctement configurés et que les mesures de sécurité appropriées sont mises en œuvre, ou bien qu'ils soient désactivés s'ils ne sont pas nécessaires.

Un autre aspect important est l'évaluation de la robustesse des mots de passe utilisés pour accéder aux équipements. Des mots de passe faibles ou faciles à deviner peuvent être exploités pour accéder aux systèmes et aux données sensibles. Il est donc crucial de vérifier si les mots de passe sont suffisamment complexes, régulièrement mis à jour et correctement protégés.

En outre, lors des tests de configuration, on vérifie également si les équipements sont régulièrement mis à jour avec les derniers correctifs de sécurité. Les mises à jour logicielles permettent de corriger les vulnérabilités connues et de renforcer la sécurité des systèmes.

Enfin, il est important d'identifier et de corriger les configurations par défaut non sécurisées. De nombreux équipements sont livrés avec des paramètres par défaut qui peuvent présenter des risques de

sécurité. L'audit de configuration permet de détecter ces configurations par défaut et de les modifier selon les bonnes pratiques de sécurité.

L'ensemble de ces tests de configuration vise à identifier les vulnérabilités liées à la configuration des équipements et à recommander des mesures correctives pour renforcer la sécurité. Les résultats de ces tests sont documentés et inclus dans le rapport final, fournissant ainsi à le fournisseur une évaluation précise de la conformité des équipements aux exigences de sécurité du projet.

## 3.7 BeStorm

BeStorm est l'outil que nous utilisons pour effectuer des tests de fuzzing. Le fuzzing est une technique utilisée pour tester la résilience et la stabilité des applications logicielles. Elle consiste à envoyer des entrées de données aléatoires, incorrectes ou inattendues à une application dans le but de provoquer des erreurs, des plantages ou des comportements non désirés.

L'objectif principal du fuzzing est de détecter des vulnérabilités, telles que des failles de sécurité ou des erreurs de programmation, en exposant les applications à des conditions auxquelles elles n'ont pas été correctement préparées. En soumettant des données d'entrée non conformes, le fuzzing permet d'explorer les limites et les faiblesses potentielles d'une application.

Le processus de fuzzing peut être effectué de manière automatisée en utilisant des outils spécialement conçus, qui génèrent et envoient automatiquement des entrées de données variées à l'application cible. Ces entrées peuvent inclure des chaînes de caractères, des fichiers, des requêtes réseau, etc.

Le fuzzing peut être appliqué à différents niveaux de l'application, tels que l'interface utilisateur, les protocoles réseau, les fichiers d'entrée, les API, etc. Il permet de mettre en évidence des erreurs de validation, des dépassements de tampon, des fuites d'informations, des erreurs de manipulation des données, et d'autres vulnérabilités qui pourraient être exploitées par des attaquants.

Je n'ai pas eu l'occasion d'effectuer de fuzzing. Cependant, il est important de comprendre l'objectif et l'utilité de cette technique car elle est pratiquée lors des COVASEC finaux.

# 4 Veille cyber

## 4.1 Rôle de la veille cyber

La veille cyber occupe un rôle critique dans le contexte d'une grande entreprise telle que la SNCF. Elle consiste à surveiller en permanence les évolutions, les tendances et les menaces dans le domaine de la cybersécurité afin de protéger les infrastructures et les systèmes contre les attaques potentielles. Par exemple, la veille cyber permet de détecter les nouvelles techniques d'attaque telles que les ransomwares ou les attaques par hameçonnage utilisées par les cybercriminels. Elle permet également de rester à jour sur les vulnérabilités émergentes dans les logiciels et les protocoles utilisés par la SNCF, comme les failles de sécurité dans les systèmes de contrôle des trains ou dans les applications de réservation en ligne. En surveillant activement ces informations, l'entreprise peut prendre des mesures préventives pour renforcer sa sécurité et réduire les risques de cyberattaques.

La veille cyber joue un rôle clé dans la gestion des incidents de sécurité. En surveillant les bulletins de sécurité, les rapports d'incidents et les forums spécialisés, la SNCF peut être rapidement informée des nouvelles menaces ou des vulnérabilités exploitées par des attaquants. Par exemple, si une vulnérabilité critique est découverte dans un logiciel largement utilisé par la SNCF, la veille cyber permettra de réagir rapidement en appliquant les correctifs appropriés pour prévenir toute exploitation de cette vulnérabilité. De plus, en suivant les bonnes pratiques recommandées par les organismes de sécurité,

tels que l'Agence nationale de la sécurité des systèmes d'information (**ANSSI**), la SNCF peut renforcer sa posture de sécurité et assurer la continuité de ses services essentiels.

En somme, la veille cyber est d'une importance cruciale pour l'entreprise. Elle permet de détecter les nouvelles menaces et vulnérabilités, de prévenir les attaques potentielles et de réagir efficacement en cas d'incident de sécurité. En restant à jour sur les développements de la cybersécurité, la SNCF peut protéger ses infrastructures ferroviaires, préserver la confidentialité des données des utilisateurs et garantir la fiabilité de ses services de transport.

## 4.2 Base de données de versions logiciels et protocoles

Mon rôle dans la veille cyber a été de mettre en place un tableau répertoriant les services fréquemment identifiés lors de nos scans. Pour chaque service, j'ai recherché les sources où les dates de sortie des différentes versions sont renseignées. Cette démarche nous permet d'évaluer si ces versions sont obsolètes ou trop anciennes, en tenant compte des années à venir et des vulnérabilités connues.

Le service de cybersécurité industrielle auquel j'appartiens a fixé une limite de 5 ans pour considérer qu'un service est acceptable. Au-delà de cette période, le rapport indiquera un point négatif, soulignant la nécessité de mettre à jour ces services pour garantir une sécurité optimale.

Cette veille cyber est essentielle pour anticiper les potentielles failles de sécurité liées à des versions obsolètes. Elle nous permet de prendre des mesures préventives en identifiant les services qui nécessitent une mise à jour ou une mise à niveau.

# 5 Rapport COVASEC

## 5.1 Analyse des résultats des scans et tests

Une fois tous nos tests terminés, nous devons remettre la synthèse de nos résultats au projet ainsi qu'à le fournisseur, ce document est appelé « rapport COVASEC ». Ce rapport est composé de données spécifiques au projet, ce qui permet de replacer le rapport dans son contexte en fournissant des informations sur les équipements, leur rôle et leur architecture.

Ensuite, nous résumons étape par étape les analyses que nous avons effectuées pour chaque équipement. Pour chaque résultat, nous présentons nos constatations, nos commentaires et donnons un avis. Ces avis sont regroupés en quatre catégories :

- OK : indiquant que le test est concluant et que tout est en ordre pour la mise en service.
- OKR (OK avec Réserve) : signifiant que le test est OK, mais avec des réserves. Ces réserves ne bloquent pas la mise en service immédiate, mais il est important de les prendre en compte lors d'une prochaine évolution de l'équipement concerné.
- NOK (Non OK) : démontrant que le test n'est pas concluant et nécessite des essais supplémentaires avant la mise en service.
- N/A (Non applicable) : utilisé lorsque le test ou l'évaluation n'est pas applicable à l'équipement en question.

Ce format de rapport clair et structuré permet à le fournisseur de comprendre rapidement nos résultats, nos observations et nos recommandations, facilitant ainsi la prise de décision quant à la mise en service du projet.

## 5.2 Evaluation finale et avis sur la mise en service des projets

J'ai eu l'opportunité de travailler sur un projet d'envergure où j'ai été chargé de rédiger le rapport complet de l'un des sous-systèmes.

La rédaction de ce rapport m'a permis de mener ce COVASEC jusqu'à son terme de manière autonome, tout en me faisant prendre conscience des limites acceptables en matière de risques de cybersécurité. Cela m'a également amené à poser des questions auxquelles je n'aurais pas pensé quant au fonctionnement ultérieur du projet. En effet, dès lors qu'il y a des constatations négatives dans nos rapports, il est nécessaire d'établir une nouvelle date pour de nouveaux tests jusqu'à ce que l'équipe ne détecte plus aucune faille.

De plus, étant donné que nos outils sont régulièrement mis à jour, il peut arriver qu'entre deux versions de test, de nouvelles vulnérabilités soient identifiées, même si elles n'avaient pas été détectées précédemment. Pour la rédaction de ce rapport, j'ai participé activement, au même titre que les autres membres de l'équipe, aux réunions avec le fournisseur. Cela nous a permis de confronter nos résultats et leurs justifications, que nous avons pris en compte dans la rédaction de nos rapports.

En somme, cette expérience m'a permis de comprendre l'importance de la communication avec le fournisseur, de la mise à jour constante de nos connaissances et de l'adaptation continue de nos méthodologies pour garantir la sécurité du projet.

# 6 Problèmes rencontrés

Nous avons précédemment observé un résultat lié au protocole SNMPv1 lors de notre analyse Nmap. Suite à cette découverte, nous avons poursuivi nos tests jusqu'à atteindre la phase de test de compromission.

Cependant quelque chose d'étrange s'est produit, Metasploit a généré des correspondances pour le protocole SNMP, mais les résultats obtenus étaient dépourvus de sens. En effet, Metasploit a signalé la présence de noms de communauté valides et prétendu avoir réussi à s'y connecter, ainsi que des couples d'identifiants et de mots de passe. Cependant, cela n'a pas de sens pour le protocole SNMPv1, car il ne prend pas en charge l'authentification. Cette anomalie soulève des questions.

En me rappelant des cours dispensés à l'IUT sur le protocole SNMP, j'ai proposé d'établir une connexion directe en usurpant l'identité d'un équipement légitime afin d'explorer les informations qu'un SNMP-get pourrait nous fournir. L'idée était de tester les noms de communauté valides identifiés par Metasploit pour voir s'ils étaient effectivement fonctionnels et permettaient d'accéder à des informations sensibles via le protocole SNMP.

Cependant, il nous a été impossible d'établir une connexion au SNMP, ce qui suggère un dysfonctionnement soit au niveau de la configuration, soit en raison d'un problème d'accès. D'après mes observations, il est probable que le problème soit lié à l'une de ces deux causes.

Pour approfondir l'analyse sur le SNMP, j'ai lancé plusieurs scripts spécifiques sur ce port avec nmap. Cette approche m'a permis de découvrir également des noms de communauté valides.

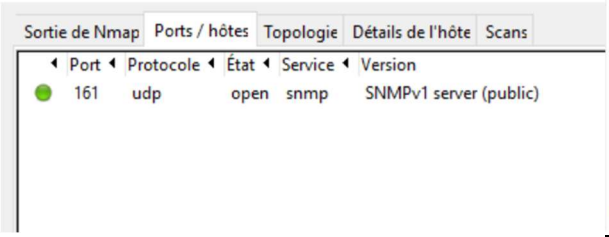


Figure 16 - Scan Nmap sans script

```
PORT    STATE SERVICE VERSION
161/udp open  snmp    SNMPv1 server (public)
| snmp-brute:
|   \x02\x02Ji\x02\x03 - Valid credentials
|   cascade - Valid credentials
|   secret - Valid credentials
|   rmonmgmtuicomm - Valid credentials
|   ANYCOM - Valid credentials
|   TENmanUFactOry - Valid credentials
|   ILMI - Valid credentials
|   MiniAP - Valid credentials
|   PRIVATE - Valid credentials
|   admin - Valid credentials
|   private - Valid credentials
|   public - Valid credentials
|   PUBLIC - Valid credentials
|   snmpd - Valid credentials
|   cisco - Valid credentials
|   mngt - Valid credentials
|_  snmp-Trap - Valid credentials
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop
```

Figure 17 - Scan Nmap avec script

Lorsque l'outil Nmap est utilisé à son plein potentiel, nous pouvons constater une nette différence dans les résultats.

Dans un dernier effort, j'ai utilisé l'outil Hydra ainsi que la console Metasploit en ligne de commande directement pour mener une attaque de type bruteforce sur le protocole SNMP en utilisant la même configuration d'équipement légitime. Hydra est un outil de test de pénétration qui permet d'automatiser des attaques par force brute sur différents protocoles. Lors de cette attaque, tous les noms de communauté proposés par Hydra se sont avérés valides, ce qui confirme l'existence d'un problème sous-jacent.

```
msf6 > use auxiliary/scanner/snmp/snmp_login
msf6 auxiliary(scanner/snmp/snmp_login) > set RHOSTS 10.83.0.1
RHOSTS ⇒ 10.83.0.1
msf6 auxiliary(scanner/snmp/snmp_login) > run

[!] No active DB -- Credential data will not be saved!
[+] 10.83.0.1:161 - Login Successful: seri (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: security (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: secret (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: scotty (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: publ1c (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: proxy (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: pr1v4t3 (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: password (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: yellow (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: xyzzy (Access level: read-only); Proof (sysDescr.0):
[+] 10.83.0.1:161 - Login Successful: write (Access level: read-only); Proof (sysDescr.0):
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Figure 18 - Console de commande Metasploit

À ce jour, je n'ai toujours pas obtenu de réponse exacte quant à la cause de ce problème. Malgré mes investigations approfondies et mes différentes tentatives d'analyse, il reste encore des zones d'ombre concernant les raisons précises qui ont conduit à ces comportements anormaux du protocole SNMP. Le problème a été soulevé au fournisseur qui est en cours d'investigation sur le sujet.

# 7 Conclusion

En somme, ce rapport de stage a mis en évidence mon expérience enrichissante au sein du département d'Ingénierie & Services en Cyber Sécurité industrielle. J'ai eu l'opportunité de comprendre les enjeux spécifiques de ce domaine en apprenant à intégrer la cybersécurité dans les projets, à effectuer des tests de vulnérabilité et de compromission. Grâce à la méthodologie de travail rigoureuse que j'ai suivie, j'ai pu acquérir des compétences pratiques dans l'utilisation d'outils tels que Nmap, Besecure, Metasploit et AppSpider.

L'importance des bases solides en réseaux que j'ai acquises à l'IUT ne peut être sous-estimée. Elles ont été indispensables pour mener à bien ce stage.

En conclusion, le COVASEC a été réalisé avec succès, avec peu d'obstacles à l'exception d'une légère incompréhension. Je reste attentive à la résolution de cette question afin de ne pas rester dans l'ignorance pour les futurs projets similaires que je pourrais être amenée à rencontrer. Tant ma formation académique que mon stage ont été extrêmement bénéfiques pour mon parcours, et je prévois de poursuivre dans le domaine de la cybersécurité dès l'année prochaine en continuant en alternance.

Au fil des années, mon objectif initial était de me spécialiser dans la sécurité offensive. Cependant, à mesure que j'approfondis mes connaissances et que je pratique, je suis de plus en plus attirée par la sécurité défensive, car nous en avons cruellement besoin.

Je suis très enthousiaste pour la suite.

# 8 Remerciements

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué à la réalisation de ce rapport de stage.

Tout d'abord, je souhaite remercier Valentine, qui a joué un rôle essentiel en me mettant en relation avec l'équipe et en facilitant la mise en place de ce stage. Sans son soutien et son dévouement, cette opportunité n'aurait pas vu le jour.

Un grand merci également à Yseult, qui a accepté de m'accueillir au sein de son département pour réaliser ce stage. Sa confiance en moi et son ouverture d'esprit ont été des éléments clés dans la réussite de cette expérience professionnelle.

Je tiens également à exprimer ma gratitude envers Eva et Chamsoudine, qui m'ont chaleureusement accueillie au sein de leur équipe. Leur accompagnement bienveillant, leur disponibilité et leur intégration rapide m'ont permis de me sentir pleinement intégrée dès les premiers jours.

Un remerciement spécial à Nicolas, dont l'aide précieuse et les conseils éclairés ont été d'une grande valeur. Son expérience m’a permis de découvrir la dimension internationale de ce domaine, qui correspond à mes aspirations professionnelles.

Je tiens également à exprimer ma profonde gratitude envers Laurent, sa bonne humeur contagieuse et son attitude positive ont eu un impact significatif sur ma motivation et mon intégration au sein de l'entreprise.

Enfin, je tiens à exprimer ma sincère reconnaissance envers Julie, ma tutrice, ainsi que Christophe, avec qui j'ai réalisé tous mes tests. Leur patience, leur pédagogie et leur soutien constant ont été d'une aide inestimable tout au long de cette découverte du monde professionnel en cybersécurité.

Je tiens également à exprimer ma profonde gratitude envers l'équipe de Lyon qui m'a chaleureusement accueillie lors de ma journée de découverte du cloud. Leur gentillesse, leur expertise et leur volonté de partager leurs connaissances ont été très enrichissantes.

Enfin, je souhaite exprimer ma reconnaissance envers l'entreprise qui m'a offert cette opportunité de stage et qui a contribué à mon développement professionnel.

Merci à tous pour votre soutien, votre guidance et votre confiance tout au long de ce stage. Votre contribution a été essentielle à ma réussite et à mon épanouissement dans le domaine de la cybersécurité.

# 9 Glossaire

**Drapeau TCP,** Un drapeau TCP est un bit dans l'en-tête TCP qui indique un état ou une action spécifique lors d'une communication TCP, tels que l'établissement, la confirmation ou la fermeture de la connexion.

**Payload**, un payload est un morceau de code qui est injecté dans une cible vulnérable afin d'exécuter une action spécifique, telle qu'une ouverture de session ou une prise de contrôle à distance.

**Module,** un module est une pièce de code qui permet d'automatiser une tâche ou une séquence d'actions spécifiques dans le processus d'exploitation d'une vulnérabilité. Les modules peuvent inclure des exploits, des scanners, des auxiliaires, etc.

**Exploit,** un exploit est un module spécifique qui exploite une vulnérabilité connue dans un logiciel ou un système cible. Il permet de profiter de la faille de sécurité pour accéder à distance à la cible, exécuter du code malveillant ou effectuer d'autres actions selon la nature de la vulnérabilité

# 10 Sitographie


https://www.sncf.com/fr/groupe/patrimoine/deux-siecles-histoire

https://fr.wikipedia.org/wiki/Histoire_de_la_SNCF

https://numerique.sncf.com/actualites/cybersecurite-un-positionnement-strategique-de-la-transformation-numerique-de-sncf/

https://fr.wikipedia.org/wiki/User_Datagram_Protocol

https://fr.wikipedia.org/wiki/Transmission_Control_Protocol

https://www.metasploit.com/

https://docs.rapid7.com/appspider/