Tests d'intrusion boîte noire de machines virtuelles

Alexis Brouste
Tàzio Gennuso

ENSIIE

–

SEC2

6 décembre 2018

Table des matières

Préambule

Le présent rapport décrit et résume les tests d'intrusion réalisés sur quatre machines virtuelles pour le projet du cours de SEC2.

Un test d'intrusion consiste à découvrir des vulnérabilités sur le système d'information audité et à vérifier leur exploitabilité et leur impact, dans les conditions réelles d'une attaque, à la place d'un attaquant potentiel.

Les menaces identifiées ont alors leur criticité qualifiée et un score leur est attribué selon l'impact et la facilité d'exploitation.

Les tests d'intrusion ont été réalisés en « boîte noire », ce qui signifie que l'auditeur a réalisé les tests sans accès aux machines et sans connaissance des systèmes. Une première phase de reconnaissance et d'analyse a donc été nécessaire pour les identifier et les pénétrer.

Les machines virtuelles avaient pour but de simuler des systèmes vraisemblables (dans la limite pédagogique) et responsables de l'hébergement d'un service, le plus souvent des sites web. Quelle que soit la sensibilité des données que ces systèmes auraient pu traiter, leur infiltration permet de perturber la disponibilité et l'intégrité, qui sont des besoins en sécurité majeurs.

Les objectifs des tests de pénétration étaient pédagogiques et consistaient à obtenir des accès super-utilisateur sur les machines pour récupérer des *drapeaux*, preuves du succès. Pour chaque machine, la méthode consistait à trouver une entrée, puis d'exploiter une faille permettant d'obtenir les privilèges requis pour lire le drapeau, simulation des méthodes d'attaques classiques par des personnes malveillantes.

Le première machine virtuelle a été examinée en profondeur dans le but de se conformer à un audit de sécurité pour se familiariser avec l'exercice, tandis que l'analyse des trois suivantes s'est concentrée sur la recherche de vulnérabilités à exploiter afin de trouver les drapeaux. Leur audit sera donc moins exhaustif.

1. Première machine virtuelle

1.1 Synthèse

Le test d'intrusion de type « boîte noire » visant la première machine virtuelle a permis de mettre en évidence vingt-et-une vulnérabilités dont la criticité varie de mineure à critique. Les critères de disponibilité, d'intégrité et de confidentialité sont concernés par ces vulnérabilités. Le niveau de sécurité de la machine virtuelle considéré comme ***très faible*** au regard du niveau de criticité et de l'impact des vulnérabilités découvertes.

L'application des recommandations pour les vulnérabilités majeures et critiques permettrait de rapidement atteindre un niveau de sécurité considéré comme ***correct***. La machine semble très ancienne et obsolète, aussi certaines recommandations impliquent des actions conséquentes voire complexes, et nous estimons qu'il serait préférable de migrer le service web sur une machine plus récente afin de partir d'une base saine, plutôt que de tenter de consolider la machine actuelle.

L'exploitation des vulnérabilités majeures et critiques mettent fortement en péril la sécurité de la machine. En exploitant ces vulnérabilités, un attaquant pourrait notamment parvenir à :

- rendre indisponible l'application ;
- télécharger la totalité des fichiers de la machine, comprenant le code source des applications les fichiers de configuration contenant des informations critiques ;
- prendre le contrôle total du système d'exploitation.

Il est cependant important de préciser que la réussite de ces attaques repose sur l'exploitation d'un faible nombre de vulnérabilités et que de nombreuses ont été omises au vu de la vétusté du système actuel.

1.1.1 Vulnérabilités observées

Identifiant	Criticité	Description de la vulnérabilité
V00	Important	Service d'accès à distance accessible
V01	Majeur	Faille permettant de vérifier si un utilisateur local existe
V02	Mineur	*Directory listing* activé
V03	Critique	Système hôte obsolète
V04	Critique	Interception de flux par attaque *man-in-the-middle*
V05	Mineur	Méthode HTTP *TRACE* activée
V06	Important	Certificat de sécurité auto-signé
V07	Majeur	Certificat de sécurité expiré
V08	Majeur	Algorithmes cryptographiques obsolètes et dépréciés
V09	Important	Autorités de certification non sûres
V10	Important	Trafic web en clair
V11	Important	Vulnérabilité sur fuite de cookies
V12	Important	Faible sécurité cryptographique des certificats
V13	Mineur	Serveur de base de données exposé
V14	Majeur	Faille de type injection SQL
V15	Majeur	Exécution de commandes depuis l'interface d'administration
V16	Important	Pas de pare-feu sur la machine
V17	Majeur	Possibilité de découverte réseau autour de la machine
V18	Mineur	Positionner l'en-tête X-Frame-Option dans toutes les réponses
V19	Mineur	Vulnérabilité de type « Protection XSS du navigateur non activée »
V20	Mineur	Vulnérabilité de type « Protection XSS du navigateur non activée »

TABLE *1.1 – Récapitulatif des vulnérabilités*

1.1.2 Recommandations proposées

Identifiant	Vulnérabilités couvertes	Description de la recommandation	Priorité de traitement	Difficulté de mise en œuvre
R00	V01	Mettre à jour le service d'accès à distance	Élevée	Facile
R01	V02 V05 V10	Durcir la configuration du serveur web	Faible	Faible
R02	V03	Mettre à jour le système d'exploitation	Élevée	Difficile
R03	V04 V08 V11	Mise à jour de la bibliothèque cryptographique et du service web	Élevée	Moyenne
R04	V06 V07 V09 V12	Génération de nouveau certificat de sécurité et signature par un organisme reconnu	Élevée	Moyenne
R05	V13	Modification de la configuration du service SQL pour utiliser des *sockets* locales	Moyenne	Facile
R06	V14 V15	Ajout de fonctions d'aseptisation sur les entrées utilisateurs de l'application web	Élevée	Facile
R07	V00 V16 V13	Installer et configurer un pare-feu	Élevée	Moyenne
R08	V17	Supprimer la fonctionnalité non-nécessaire ou configurer un pare-feu	Moyenne	Moyenne
R09	V18	Positionner l'en-tête X-Frame-Option dans toutes les réponses	Faible	Facile
R10	V19	Positionner l'en-tête X-XSS-Protection dans toutes les réponses	Faible	Facile
R11	V20	Positionner l'en-tête X-Content-Type-Options dans toutes les réponses	Faible	Facile

TABLE *1.2 – Récapitulatif des recommandations*

Les vulnérabilités ainsi que les recommandations seront détaillées dans la suite du rapport, mais chacun ne l'est qu'une fois. un recommandation peut ainsi s'appliquer à plusieurs vulnérabilités, aussi il est préférable de se référer au tableau précédent pour savoir quelle recommandation appliquer face à chaque vulnérabilité.

1.2 Tests d'intrusion

1.2.1 Reconnaissance passive

La cible étant une machine virtuelle isolée, la seule méthode de reconnaissance passive était l'interrogation de l'hyperviseur pour savoir si la machine virtuelle avait bien obtenue une IP dynamique sur le réseau interne, ce qui fut le cas.

La machine était opérationnelle et prête à être analysée.

1.2.2 Reconnaissance active

1.2.2.1 Identification du système d'exploitation

```
~$ sudo nmap -O 192.168.122.204
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-05 19:17 CET
Nmap scan report for 192.168.122.204
Host is up (0.00037s latency).
Not shown: 994 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
111/tcp  open  rpcbind
443/tcp  open  https
631/tcp  open  ipp
3306/tcp open  mysql
MAC Address: 52:54:00:E2:2F:E8 (QEMU virtual NIC)
Device type: general purpose|media device
Running: Linux 2.6.X, Star Track embedded
OS CPE: cpe:/o:linux:linux_kernel:2.6 cpe:/o:linux:linux_kernel:2.6.23 cpe:/h:star_track:srt2014hd
OS details: Linux 2.6.9 - 2.6.30, Star Track SRT2014HD satellite receiver (Linux 2.6.23)
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.45 seconds
```

FIGURE *1.1 – Identification de système d'exploitation et scan de ports ouverts sur le système cible*

Afin de mieux cibler et connaître le système cible, la première étape incontournable est d'essayer d'identifier le système d'exploitation sur la machine cible. Grâce à l'outil `nmap` analysant le comportement de la pile réseau, cela peut se faire aisément. Il se peut cependant que certains systèmes soient volontairement muets face à ce type de scan, mais ce n'est pas le cas ici, et nous avons une sortie qui nous indique que le système est à base de Linux, en version entre la 2.6.9 et la 2.6.30.

Les informations sur le type d'utilisation du système, comme récepteur satellite s'avèrent fausses, mais donnent une information correcte, le Linux en question est bien en version 2.6.9 comme on le verra par la suite.

La version 2.6.9 du noyau est périmée depuis plus de treize ans, ce qui est un gigantesque problème de sécurité puisque de nombreuses failles ont été révélées depuis lors sur les versions obsolètes de Linux. Ces résultats sont une fuite d'information, qui conjuguées à une vulnérabilité éventuelle sur un des services tournant sur la machine, peuvent offrir à un attaquant une porte pour obtenir le contrôle complet à travers une vulnérabilité du noyau.

La sécurité du système n'est de ce fait pas assurée.

Vulnérabilité 3	**Système hôte obsolète**	**CVSS 10/10**
La machine fonctionne sous un vieux noyau Linux âgé de plus de treize ans, de nombreuses failles faciles à mettre en œuvre sont éprouvées pour obtenir les accès super-utilisateur.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Moyenne	Critique

Recommandation 2	Mettre à jour le système
Mettre à jour le système et *de facto* les services sur la machine (risque de cassage de configuration), ou bien transférer le site web vers un hôte plus à jour.	
Priorité de traitement	**Difficulté de mise en œuvre**
Haute	Difficile

1.2.2.2 Identification des services

Le premier scan nous donne des informations sur les ports ouverts et sur les ports fermés. On peut en déduire que la machine ne possède pas de pare-feu, ce qui est en soi une vulnérabilité.

Vulnérabilité 16	Aucun pare-feu installé	CVSS 6.5/10
Pas de pare-feu installé sur la machine, ce qui laisse de la marge de manœuvre à un attaquant pour utiliser les flux qu'il souhaite. En outre, cela laisse la porte ouverte à des services qui auraient pu être masqués de l'extérieur.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

Recommandation 7	Installer et configurer un pare-feu
Installer et configurer un pare-feu tel que `iptables` pour bloquer tous les flux non nécessaires au fonctionnement de l'application et pour bloquer l'accès depuis l'extérieur à certains services.	
Priorité de traitement	**Difficulté de mise en œuvre**
Haute	Moyenne

On observe six services fonctionnant sur la cible, dont SSH, un service d'administration à distance (chiffré, et avec authentification). Laisser un tel service accessible publiquement est un manquement à la sécurité du système.

```
user@Kali:~$ nmap 192.168.122.204 -sV
Starting Nmap 7.70 ( https://nmap.org ) at 2018-11-23 10:01 CET
Nmap scan report for 192.168.122.204
Host is up (0.0015s latency).
Not shown: 994 closed ports
PORT     STATE SERVICE    VERSION
22/tcp   open  ssh        OpenSSH 3.9p1 (protocol 1.99)
80/tcp   open  http       Apache httpd 2.0.52 ((CentOS))
111/tcp  open  rpcbind    2 (RPC #100000)
443/tcp  open  ssl/https?
631/tcp  open  ipp        CUPS 1.1
3306/tcp open  mysql      MySQL (unauthorized)
```

FIGURE *1.2 – Identification des versions des services hébergés sur la machine*

Vulnérabilité 0	Service d'accès à distance accessible	CVSS 4.5/10
Le service d'administration à distance, même s'il présente une authentification, doit être filtré et protégé d'un accès public immédiat.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

De même, un service de base de données SQL est accessible depuis l'extérieur, alors qu'il est usuel et plus sage de l'isoler du réseau. Les bonnes pratiques de sécurité ne sont pas respectées.

Vulnérabilité 13	Service de base données SQL accessible	CVSS 4.5/10
Outre le service web, les autres services doivent être filtrés pour empêcher les attaquants de directement communiquer avec eux et d'exploiter des failles éventuelles.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

1.2.3 Analyse cryptographique

Nous avons remarqué dans un premier temps que le serveur offrait le site en HTTP, c'est-à-dire que tout le trafic entre le serveur et l'utilisateur n'est pas chiffré. Un attaquant passif peut donc, s'il écoute le réseau entre l'utilisateur

Recommandation 5	Modification de la configuration du service
Modifier la configuration du service afin qu'il écoute uniquement sur l'interface de bouclage, ou via des *sockets* Unix. Dans le cas d'un serveur non-dédié à l'hébergement d'une base de données, les services ayant besoin d'accéder à SQL tournent sur la machine locale et peuvent y accéder par le biais de mécanismes locaux, inutile de l'ouvrir à l'extérieur en TCP.	
Priorité de traitement	**Difficulté de mise en œuvre**
Moyenne	Facile

et le serveur, capturer toutes les données émises par chacun d'eux. Proposer du HTTP en parallèle du HTTPS, sans redirection obligatoire pour un site contenant une partie sécurisée, ne respecte pas les critères de sécurité.

Vulnérabilité 10	Trafic web en clair, absence de chiffrement	CVSS 6/10
Le site est disponible via deux biais, HTTP et HTTPS. HTTP n'est pas un protocole sécurisé ; il faut forcer l'utilisation de HTTPS.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

Recommandation 1	Durcir la configuration du serveur web
Bien configuré, le serveur permet de réduire la surface d'attaque. Force l'utilisation seule du HTTPS, désactiver la méthode TRACE et désactiver l'affichage des répertoires (*directory listing*) sont des bonnes pratiques pour durcir un site web.	
Priorité de traitement	**Difficulté de mise en œuvre**
Moyenne	Facile

En analysant les informations cryptographiques offertes par le serveur web en mode sécurisé, nous avons découvert plusieurs vulnérabilités, notamment l'utilisation de suites cryptographiques et algorithmes obsolètes et dépréciés, ce qui permet à un attaquant de casser le chiffrement entre un utilisateur et le serveur. De fait, le critère de sécurité n'est pas respecté.

Vulnérabilité 8	Algorithmes cryptographiques obsolètes et dépréciés	CVSS 5/10
Des vulnérabilités connues sont présentes sur les algorithmes cryptographiques utilisés par le serveur web.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

Recommandation 3	Mise à jour des bibliothèques cryptographiques et du serveur web
Les bibliothèques cryptographiques contiennent les algorithmes de chiffrement. Mises à jour, cela permet le retrait et la correction d'algorithmes vulnérables. Pour être compatible, le serveur web ainsi que sa configuration doivent être mis à jour pour forcer l'utilisation d'algorithmes plus récents.	
Priorité de traitement	**Difficulté de mise en œuvre**
Élevée	Moyenne

L'analyse du certificat émis par le serveur pour la connexion sécurisée a elle aussi révélé des failles. Notamment au niveau des algorithmes utilisés pour sa confection. Le certificat a de plus été signé par une autorité de certification inconnue, ce qui ne permet pas d'assurer son authenticité par l'utilisateur. De plus le certificat est périmé.

La connexion sécurisée entre le serveur et le client est donc très facile à compromettre.

Vulnérabilité 6	Certificat de sécurité auto-signé	CVSS 5/10
L'utilisateur ne peut pas s'assurer de l'identité du site web.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

Recommandation 4	Renouvellement du certificat de sécurité du serveur web
Le certificat permet d'assurer que le site visité par l'utilisateur est le bon et ainsi établir une connexion sûre. Lorsque le certificat est expiré ou signé par une autorité inconnue, il devient possible d'usurper l'identité du site. Générer un nouveau certificat correctement signé permet une sécurité accrue pour les utilisateurs.	
Priorité de traitement	**Difficulté de mise en œuvre**
Élevée	Facile

Vulnérabilité 7	Certificat de sécurité expiré	CVSS 5/10
L'utilisateur peut se retrouver sur un site compromis par mégarde, lequel aurait usurpé l'identité du site web originel.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

Vulnérabilité 9	Autorité de certification inconnue	CVSS 5/10
L'utilisateur ne peut pas s'assurer de l'identité du site web et peut être soumis à la malfaçon.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

1.2.4 Scan de vulnérabilités

1.2.4.1 Scan de vulnérabilités système

Afin d'identifier les vulnérabilités touchant la cible ainsi que ses services, le scanneur de vulnérabilités OpenVas a été utilisé.

OpenVas remonte quatre nouvelles vulnérabilités, deux touchant le serveur web et sa configuration et deux sur la sécurité des connexions sécurisées. La configuration du serveur web actuelle peut causer des fuites d'information, telles que les sessions des utilisateurs. Le certificat de sécurité peut aussi être compromis, et des attaques de détournement de flux sécurisés pour déchiffrement sont possibles. Il est recommandé de durcir le système.

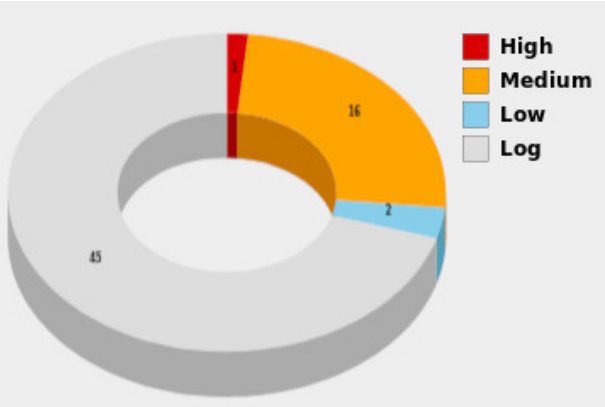

FIGURE *1.3 – Rapport de vulnérabilités par OpenVas*

Vulnérabilité 4	Interception de flux par attaque *man-in-the-middle*	CVSS 6.8/10
Des vulnérabilités sont présentes dans les bibliothèques cryptographiques utilisées par le serveur web pour les connexions sécurisées qui permettent aux attaquants de déchiffrer le trafic dans certains cas.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Moyenne	Majeur

Vulnérabilité 11	Fuite de cookie	CVSS 4.3/10
La configuration du serveur web ne met pas en place une sécurité sur la divulgation de cookie, ce qui rend les failles de type « XSS » possibles et risque de répandre des informations utilisateurs.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Moyenne	Mineur

1.2.4.2 Scan de vulnérabilités applicatives

Les scanneurs tels qu'OpenVas sont destinés à scanner un système dans son ensemble pour trouver des vulnérabilités, mais les services hébergés sur celui-ci nécessitent un scan plus spécifique par des applications dédiées. Ici

Vulnérabilité 12	Faible sécurité cryptographique du certificat de sécurité	CVSS 5/10
L'utilisateur peut être soumis à la malfaçon par redirection vers un site malicieux qu'il pense sûr.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

Vulnérabilité 5	Méthode HTTP *TRACE* activée	CVSS 3.7/10
Cette méthode HTTP est généralement utilisée pour le débogage d'applications et peut être utilisée malicieusement par un tiers pour voler les cookies de session d'un utilisateur.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Mineur

Vulnérabilité 1	Faille permettant de vérifier si un utilisateur local existe	CVSS 5.9/10
Faille dans le service d'accès à distance qui permet à un attaquant de vérifier avec brutalité si un utilisateur existe sur la machine.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Facile	Important

Recommandation 0	Mettre à jour le service d'accès à distance
Installer une version plus récente du serveur SSH qui corrige la faille.	
Priorité de traitement	**Difficulté de mise en œuvre**
Haute	Facile

`nikto` a été utilisé afin de scanner le site web à la recherche de vulnérabilités communes.

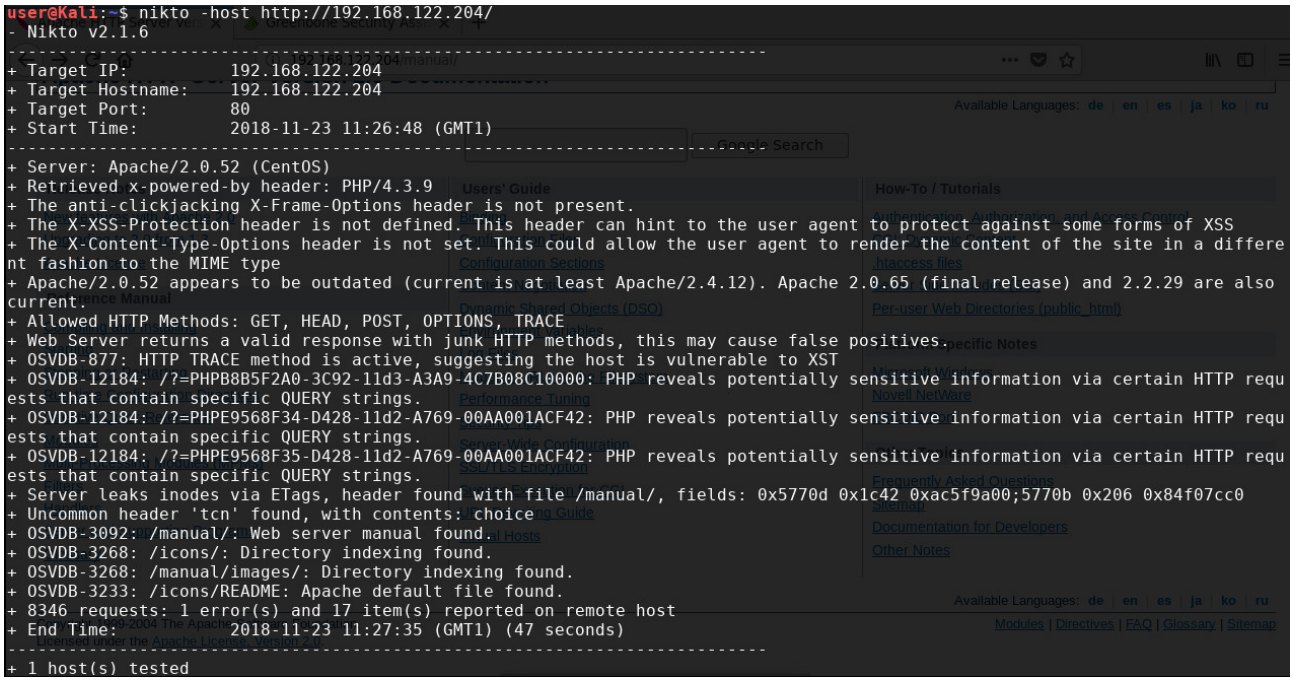

FIGURE *1.4 – Rapport de vulnérabilités web par nikto*

La scanneur révèle notamment que le *directory listing* est activé pour plusieurs dossiers sur le serveur. Dans le cas d'un oublie de la part du programmeur d'un fichier d'index, ceci peut conduire à la compromission de certains éléments du site.

Vulnérabilité 2	Directory listing activé	CVSS 3.7/10
Un utilisateur peut télécharger des fichiers cachés sur le serveur lorsqu'il tente d'accéder à un dossier et non une page. Cette vulnérabilité n'est en l'occurrence présente que sur des dossier d'impact minime.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Mineur

Le scanneur révèle plusieurs vulnérabilités avérées, néanmoins mineures. À elles seules ces vulnérabilités ne constituent pas une grande menace quant à l'intégrité du site web, mais conjuguées à d'autres éventuelles elles peuvent mener à la compromission d'informations utilisateur. Il est donc nécessaire de procéder à la correction de ces vulnérabilités.

Vulnérabilité 18	**Vulnérabilité de type « X-Frame-Option header not set »**	**CVSS 2.6/10**
Si l'application est compromise, un attaquant pourrait modifier une page de l'application en y ajoutant des éléments invisibles à l'oeil nu. En cliquant dessus, un utilisateur pourrait voir sa session usurpée ou encore avoir ses entrées clavier enregistrées.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Élevée	Mineur

Recommandation 9	**Positionner l'en-tête X-Frame-Option dans toutes les réponses**
L'ensemble des réponses HTTP doivent contenir l'en-tête X-Frame-Option avec comme valeur SAMEORIGIN ou DENY selon le contexte.	
Priorité de traitement	**Difficulté de mise en œuvre**
Faible	Facile

Vulnérabilité 19	**Vulnérabilité de type « Protection XSS du navigateur non activée »**	**CVSS 2.6/10**
Cet en-tête autorise le navigateur à déclencher ses propres mécanismes de détection et de prévention XSS. Il contribue donc à la sécurisation de l'application mais son absence ne permet pas à un attaquant de directement réaliser une attaque de type XSS.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Élevée	Mineur

Recommandation 10	**Positionner l'en-tête X-XSS-Protection dans toutes les réponses**
L'ensemble des réponses HTTP doivent contenir l'en-tête X-XSS-Protection avec comme valeur 1.	
Priorité de traitement	**Difficulté de mise en œuvre**
Faible	Facile

Vulnérabilité 20	**Vulnérabilité de type « Protection XSS du navigateur non activée »**	**CVSS 2.6/10**
L'en-tête X-Content-Type-Options contre le *sniffing* MIME n'est pas renseigné à « nosniff ». Ceci permet à de vielles versions d'Internet Explorer et de Chrome de pratiquer le sniffing MIME sur le corps de réponse, conduisant potentiellement à l'interprétation et l'affichage du contenu dans un autre type que celui déclaré.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Élevée	Mineur

Recommandation 11	**Positionner l'en-tête X-Content-Type-Options dans toutes les réponses**
L'ensemble des réponses HTTP doivent contenir l'en-tête X-Content-Type-Options avec comme valeur « nosniff ».	
Priorité de traitement	**Difficulté de mise en œuvre**
Faible	Facile

Le scan applicatif nous a donc permis de trouver des vulnérabilités mineures touchant la confidentialité des données utilisateur. Ces vulnérabilités concernaient surtout les mécanismes de protection du navigateur utilisateur. L'application des recommandations permettra de garantir qu'ils sont en place et limitera les risques de vols de données.

1.2.5 Exploitation

Une fois les vulnérabilités repérées, nous avons tenté une intrusion dans le système en simulant les méthodes d'un attaquant. Nous avons trouvé et exploité au cours de celles-ci des nouvelles vulnérabilités qui seront mentionnées tout

du long.

Nous avons commencé par accéder au service web pour commencer la pénétration au sein du système. On arrive immédiatement sur une page de connexion, et puisque nous avons vu précédemment qu'un service MySQL était lancé, le première étape était de tester des injections SQL. L'injection la plus trivialle fonctionne, en rentrant comme couple d'identifiants « root » et « ' OR 1=1 # », après avoir essayé avec le nom d'utilisateur « admin ». L'injection SQL est très basique avec un nom d'utilisateur commun, ce qui rend son exploitation très facile pour un attaquant et lui donne accès à la « console d'administration » du site. Les critères de sécurité ne sont pas respectés.

Vulnérabilité 14	**Faille de type injection SQL**	**CVSS 8.2/10**
L'utilisateur peut se connecter sans mot de passe à l'interface d'administration du site web notamment, mais peut aussi exécuter des commandes SQL arbitraires mettant en péril la disponibilité et l'intégrité des données.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Facile	Important

Recommandation 6	**Ajout de fonctions d'aseptisation sur les entrées utilisateurs de l'application web**
Les entrées utilisateur dans les champs de connexion et de l'interface d'administration ne sont pas protégés, il est nécessaire d'appliquer sur ces données des fonctions d'échappement de caractères spéciaux avant de les traiter.	
Priorité de traitement	**Difficulté de mise en œuvre**
Élevée	Facile

L'interface d'administration nous donne la possibilité de lancer un *ping* sur une machine quelconque. Cette fonctionnalité présente un risque puisqu'elle permet à un utilisateur distant de cartographier à distance le réseau interne sur lequel la machine est connectée.

FIGURE *1.5 – Interface d'administration du site web*

Vulnérabilité 17	**Possibilité de découverte réseau autour de la machine**	**CVSS 6.6/10**
L'interface d'administration permet grâce au ping de cartographier le réseau local de la machine hôte. Au vu des failles précédentes, l'accès à l'interface d'administration est aisé, ce qui permet aux attaquants d'avoir une meilleure connaissance de l'environnement de la machine et de risque de le compromettre.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Facile	Important

Recommandation 8	**Supprimer la fonctionnalité non-nécessaire ou configurer un pare-feu**
Cette fonctionnalité ne devrait pas être présente dans une interface d'administration. Si sa présence est nécessaire, il est préférable de fournir une liste prédéfinie d'hôtes sur lesquels exécuter le ping, ou bien mettre en place un pare-feu qui filtre le ping vers les IP internes.	
Priorité de traitement	**Difficulté de mise en œuvre**
Moyenne	Moyenne

On remarque lorsqu'on lance un ping vers une adresse quelconque que c'est la commande `/usr/bin/ping` qui est exécutée. Nous nous sommes donc demandés s'il était possible d'injecter des commandes arbitraires. L'injection de

commande est possible en injectant la commande voulue précédée d'un point-virgule. Les commandes sont exécutées avec l'utilisateur `apache`, utilisateur de moindre privilège, mais avec un noyau obsolète tournant sur la machine, cette vulnérabilité est grave.

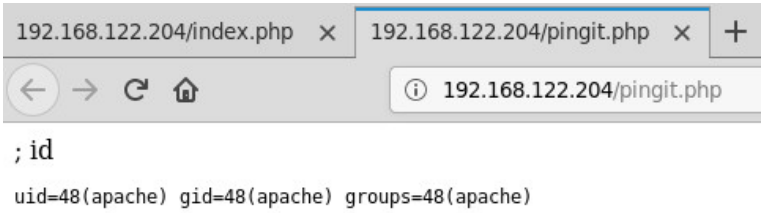

FIGURE *1.6 – Injection de la commande `id` depuis l'interface d'administration*

Vulnérabilité 15	**Exécution de commandes depuis l'interface d'administration**	**CVSS 8.3/10**
Le champ de ping de l'interface d'administration est vulnérable et permet l'exécution de commandes arbitraires sur le système hôte.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Facile	Important

Nous souhaitions obtenir un *invite de commandes inverse* afin de pouvoir tenter au mieux l'exploitation locale de la machine. Pour cela, grâce à l'exécution de `whereis`, nous avons pu déterminer que `netcat` était installé sur la machine. C'est une vulnérabilité en soi, mais nous imaginons que sa présence était purement pédagogique pour fournir une aide afin d'exploiter la première machine virtuelle, aussi nous ne le reporterons pas comme une vulnérabilité, bien que c'en soit une.

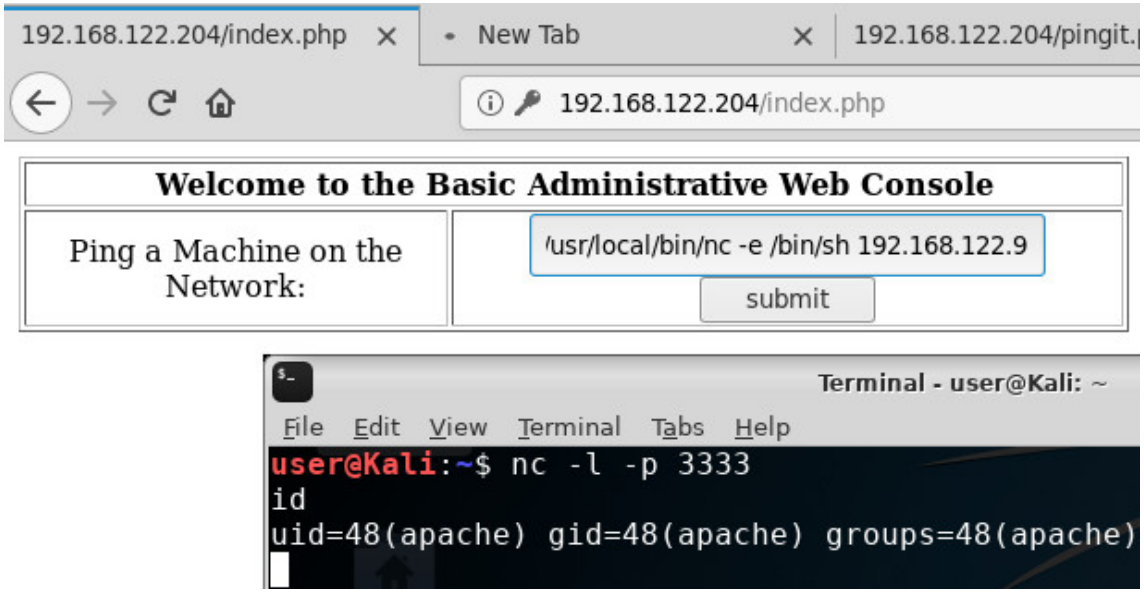

FIGURE *1.7 – Utilisation de l'injection de commande pour obtenir un invite de commandes inverse*

La capture d'écran ci-dessus montre que nous avons obtenu un accès distant sur la machine en connectant un invite de commande distant sur notre machine d'exploitation.

L'étape suivante consiste à exploiter une vulnérabilité du noyau afin d'obtenir les privilèges super-utilisateur sur la machine. Après avoir exécuté `uname` sur la machine et constaté que la version du noyau était bien la 2.6.9, nous avons recherchés quels étaient les meilleurs *exploits*. La machine était invulnérable au premier que nous avons essayé, mais au second essai, nous avons réussi grâce à `https://www.exploit-db.com/exploits/9542`. Il nous a suffi d'envoyer le fichier source sur la machine et de le compiler localement. Là encore, la présence du compilateur C gcc ne nous a as surprise en vertu de la pédagogie de l'exercice.

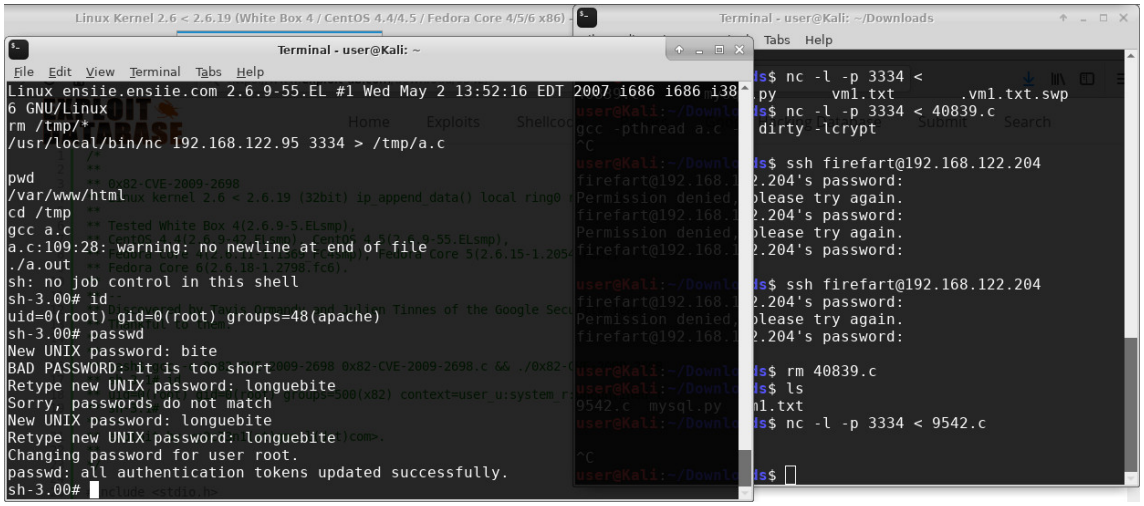

FIGURE *1.8 – Envoi de la prouesse vers la machine vulnérable et exploitation de la vulnérabilité*

Au vu de l'exploitation de la vulnérabilité et de l'obtention du l'accès super-utilisateur, la machine est fortement vulnérable.

Le code d'exploitation de la vulnérabilité permet de changer le mot de passe root ; nous sommes donc par la suite à même de nous connecter via l'outil d'administration à distance SSH.

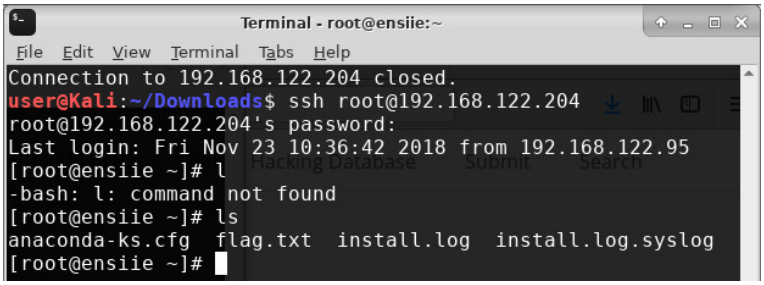

FIGURE *1.9 – Accès super-utilisateur via SSH sur la machine*

1.3 Obtention du drapeau

Une fois l'accès *root* obtenu, le jeu consistait à trouver le drapeau sur la machine. Localisé rapidement sous `/root/flag.txt`, son contenu mystérieux restait à éclaircir.

Un coup d'œil aguerri saurait reconnaître du chiffrement de César, ce qui a pu se vérifier en écrivant un simple script Python testant tous les décalages possibles. Le César en place était relativement basique puisque le décalage tenait uniquement compte de l'indice des lettres (majuscules et minuscules confondues), et non du code ASCII des caractères. Nous avons cependant été déroutés par le fait que chaque ligne utilise une clef de décalage différente. Une fois le script Python modifié, nous étions à même de récupérer le drapeau.

Après résolution de l'énigme, le premier drapeau est donc : **AZERTYUIOPQSDFGHJKLMWXCVBN**.

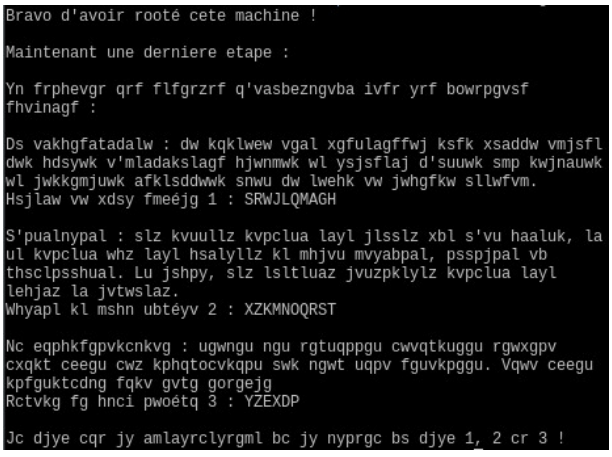

FIGURE *1.10 – Texte obscur qui prononcé tel quel permettrait selon la légende d'inviter Nyralathotep à venir manger une pizza*

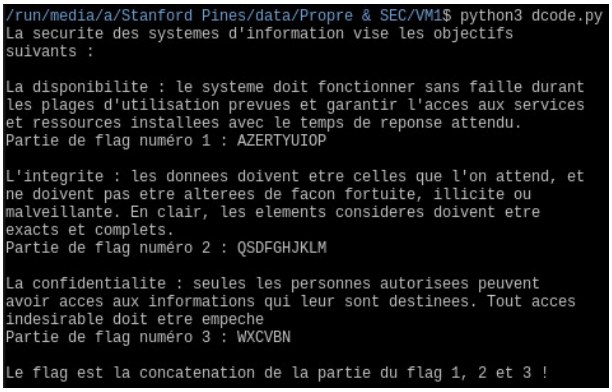

FIGURE *1.11 – Fichier flag.txt déchiffré*

2. Deuxième machine virtuelle

2.1 Synthèse

Le test d'intrusion de la deuxième machine virtuelle a permis de mettre en évidence dix-huit vulnérabilités dont la criticité varie de mineure à critique. Les critères de disponibilité, d'intégrité et de confidentialité sont concernés par ces vulnérabilités. Le niveau de sécurité de la machine virtuelle considéré comme ***faible*** au regard du niveau de criticité et de l'impact des vulnérabilités découvertes.

L'application des recommandations pour les vulnérabilités majeures et critiques permettrait de rapidement atteindre un niveau de sécurité considéré comme ***bon***. Il est important de noter qu'aucune recommandation n'implique d'actions conséquentes ou fortement complexes.

L'exploitation des vulnérabilités majeures et critiques mettent fortement en péril la sécurité de la machine. En exploitant ces vulnérabilités, un attaquant pourrait notamment parvenir à :

— rendre indisponible l'application ;
— télécharger la totalité des fichiers de la machine, comprenant le code source des applications les fichiers de configuration contenant des informations critiques ;
— prendre le contrôle total du système d'exploitation, voire des machines voisines sur le réseau interne.

Il est cependant important de préciser que la réussite de ces attaques repose sur l'exploitation d'une vulnérabilité critique, et que la corriger pourrait suffire à durcir le système.

2.1.1 Vulnérabilités observées

Identifiant	Criticité	Description de la vulnérabilité
V00	Critique	Système hôte obsolète
V01	Important	Service d'accès à distance accessible
V02	Mineur	Serveur web Apache non configuré
V03	Important	Pas de pare-feu sur la machine
V04	Critique	Faille dans Samba permettant un accès à distance privilégié

TABLE *2.1 – Récapitulatif des vulnérabilités*

2.1.2 Recommandations proposées

Identifiant	Vulnérabilités couvertes	Description de la recommandation	Priorité de traitement	Difficulté de mise en œuvre
R00	V00	Mettre à jour le système d'exploitation	Élevée	Difficile
R01	V01 V02	Filtrer les flux n'étant pas nécessaires au fonctionnement de l'application	Haute	Faible
R02	V03	Installation et configuration d'un pare-feu	Haute	Moyenne
R03	V04	Mettre à niveau Samba	Haute	Critique

TABLE *2.2 – Récapitulatif des recommandations*

2.2 Tests d'intrusion

2.2.1 Reconnaissance active

2.2.1.1 Identification du système d'exploitation

```
user@Kali:~$ nmap -sV 192.168.122.227
Starting Nmap 7.70 ( https://nmap.org ) at 2018-11-23 12:12 CET
Stats: 0:00:06 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 16.67% done; ETC: 12:12 (0:00:30 remaining)
Nmap scan report for 192.168.122.227
Host is up (0.0020s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE     VERSION
22/tcp    open  ssh         OpenSSH 2.9p2 (protocol 1.99)
80/tcp    open  http        Apache httpd 1.3.20 ((Unix)  (Red-Hat/Linux) mod_ssl
/2.8.4 OpenSSL/0.9.6b)
111/tcp   open  rpcbind     2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd (workgroup: MYGROUP)
443/tcp   open  ssl/https   Apache/1.3.20 (Unix)  (Red-Hat/Linux) mod_ssl/2.8.4
OpenSSL/0.9.6b
32768/tcp open  status      1 (RPC #100024)

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.98 seconds
```

FIGURE *2.1 – Identification de système d'exploitation et scan de ports ouverts sur le système cible*

Nous avons de prime abord scanné la machine cible afin d'identifier les services tournant dessus et la version du système d'exploitation. Nous avons identifié une machine RedHat obsolète et de fait vulnérable en autre à des attaques locales d'élévation de privilèges. Dans l'éventualité où des services vulnérables tourneraient sur la machine, des élévations de privilège sont possibles et permettraient aux attaquants de prendre le contrôle total de la machine.

La sécurité du système n'est de ce fait pas assurée.

Vulnérabilité 3	**Système hôte obsolète**	**CVSS 10/10**
La machine fonctionne sous un vieux noyau Linux âgé de plus de treize ans, de nombreuses failles faciles à mettre en œuvre sont éprouvées pour obtenir les accès super-utilisateur.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Moyenne	Critique

Recommandation 3	**Mettre à jour le système**
Mettre à jour le système et *de facto* les services sur la machine (risque de cassage de configuration), ou bien transférer le site web vers un hôte plus à jour.	
Priorité de traitement	**Difficulté de mise en œuvre**
Haute	Difficile

2.2.1.2 Identification des services

Le scan réseau grâce à `nmap` avec identification de service nous donne plusieurs services : un accès à distance SSH, un serveur web Apache et un Samba. Laisser le service SSH accessible publiquement est un manquement à la sécurité du système.

Vulnérabilité 1	**Service d'accès à distance accessible**	**CVSS 4.5/10**
Le service d'administration à distance, même s'il présente une authentification, doit être filtré et protégé d'un accès public immédiat.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

Recommandation 1	**Filtrer les flux n'étant pas nécessaires au fonctionnement de l'application**
Mettre en place ou configurer un pare-feu pour interdire l'accès depuis l'extérieur à un tel outil.	
Priorité de traitement	**Difficulté de mise en œuvre**
Haute	Moyenne

Le scan nous montre aussi à première vue un service web non sécurisé. Après visite il s'agit d'un apache déployé dans sa configuration par défaut. Laissé un service tourner avec sa configuration par défaut n'est pas sûr.

Vulnérabilité 3	**Serveur web Apache non configuré**	**CVSS 2.3/10**
Laisser un service avec sa configuration par défaut peut fournir une porte d'entrée potentielle aux attaquants.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Mineur

Nous obtenons aussi des informations sur les ports ouverts et fermés, qui traduisent une absence de filtrage. On peut en déduire que la machine ne possède pas de pare-feu, ce qui est une vulnérabilité.

Vulnérabilité 3	**Aucun pare-feu installé**	**CVSS 6.5/10**
Pas de pare-feu installé sur la machine, ce qui laisse de la marge de manœuvre à un attaquant pour utiliser les flux qu'il souhaite. En outre, cela laisse la porte ouverte à des services qui auraient pu être masqués de l'extérieur.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Difficile	Important

Recommandation 2	Installer et configurer un pare-feu
Installer et configurer un pare-feu tel que `iptables` pour bloquer tous les flux non nécessaires au fonctionnement de l'application et pour bloquer l'accès depuis l'extérieur à certains services.	
Priorité de traitement	**Difficulté de mise en œuvre**
Haute	Moyenne

2.2.2 Exploitation

La machine a très peu de surface exposée vulnérable à première vue, c'est pourquoi nous avons cherché à déterminer la version du serveur Samba hébergé. L'utilitaire `enum4linux` nous fournit une indication plus précise sur la version de Samba que netcat ; la version est dans la gamme 2.2.X, qui est obsolète.

Nous constatons qu'il existe une vulnérabilité avec une haute probabilité de succès pour ces versions, et qui permet une exploitation à distance : `https://www.exploit-db.com/exploits/10`. Une fois l'utilitaire compilé et exécuté, nous obtenons directement un invite de commande avec les privilèges super-utilisateur sur la machine. Cette vulnérabilité est très facilement exploitable puisque le code est mis en ligne à la disposition de tous. Elle offre un contrôle total à la machine pour un attaquant.

Vulnérabilité 4	Faille dans Samba permettant un accès à distance privilégié	CVSS 9.6/10
Vulnérabilité critique dans le service Samba qui offre la possibilité de prendre le contrôle total de la machine suite à l'exécution de commandes arbitraires pour un attaquant.		
Nature	**Facilité d'exploitation**	**Impact**
Technique	Facile	Critique

Recommandation 3	Mettre à niveau Samba
Il est nécessaire d'installer une version plus récente du service Samba ne présentant plus cette vulnérabilité.	
Priorité de traitement	**Difficulté de mise en œuvre**
Haute	Facile

L'accès root obtenu, la machine était entièrement sous notre contrôle.

2.3 Obtention du drapeau

Une fois en possession des pleins pouvoirs (exécutif, ~~législatif~~, etc), un fichier nommé `flag.png` nous attendait dans le répertoire personnel, pique relatée[1]. Après avoir pensé à un défi de stéganographie, et appliqué diverses méthodes basiques dans ce sens, comme le changement des couleurs, de contraste, ou l'utilisation d'outils spécifiques, nous nous sommes aperçus que le défi n'était pas là.

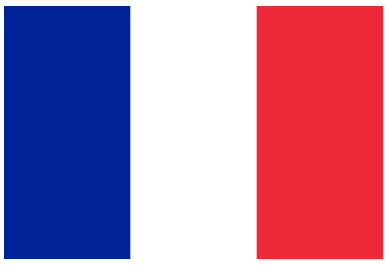

FIGURE 2.2 – *Drapeau de la ~~dictature~~ République Française trouvé sur la machine*

Nous avons donc testé la méthode consistant à lister toutes les chaînes de caractères grâce à la commande *strings*, puis nous avons eu des résultats intéressants (voir capture d'écran).

Il y a donc des messages cachés dans les métadonnées EXIF de l'image, comme nous le révèle le résultat de la commande. Il reste à savoir dans quel champ des métadonnées se trouve le bon *drapeau* (sous forme de chaîne de caractères). Grâce à la suite logicielle ImageMagick fournissant des outils d'analyse d'image, nous avons pu les visionner et obtenir le nom du champ dans lequel se trouvait le drapeau. Le drapeau final est donc **6w3D36WrkB4pqWRBNt2f89uf7X**.

1. *pic related*

```
~/wip/PROGEEG/VM2$ strings flag.png
IHDR
gAMA
 cHRM
PLTE
bKGD
tIME
xIDATx
%tEXtdate:create
2017-08-10T01:55:46+00:00<
%tEXtdate:modify
2017-08-10T01:55:46+00:00M
=tEXtComment
La fonction string n'apporte pas toujours la solution
5tEXtDescription
Le bon flag est dans le champs PNGWarning
.tEXtLabel
Le flag est : jG6fCV78qaz4R8zy3NDkU579Rkt
3tEXtDisclaimer
Le flag est : 29273usqn6gLHn7wPZK3DmdHG4
1tEXtSoftware
Le flag est : jXeQ8Bi97z956tE34weXhVHtA3
/tEXtSource
Le flag est : WVm9YT2XZg8wf5CjTee5m3f839ip6
3tEXtCollection
Le flag est : 793j59rDC8NWdyQfu8wZU2j4sW
/tEXtAuthor
Le flag est : 97sEJ5BymY28nnYDS8L5w6vsz9
0tEXtWarning
Le flag est : 6w3D36WrkB4pqWRBNt2f89uf7X#
 iTXtMake
Les metadata
a craint
IEND
```

FIGURE *2.3 – Chaînes de caractères dans l'image* `flag.png`

```
Properties:
  Author: Le flag est : 97sEJ5BymY28nnYDS8L5w6vsz9
  Collection: Le flag est : 793j59rDC8NWdyQfu8wZU2j4sW
  Comment: La fonction string n'apporte pas toujours la solution
  date:create: 2018-11-23T15:42:27+01:00
  date:modify: 2018-11-23T15:42:27+01:00
  Description: Le bon flag est dans le champs PNGWarning
  Disclaimer: Le flag est : 29273usqn6gLHn7wPZK3DmdHG4
  Label: Le flag est : jG6fCV78qaz4R8zy3NDkU579Rk
  png:bKGD: chunk was found (see Background color, above)
  png:cHRM: chunk was found (see Chromaticity, above)
  png:gAMA: gamma=0.45455 (See Gamma, above)
  png:IHDR.bit-depth-orig: 4
  png:IHDR.bit_depth: 4
  png:IHDR.color-type-orig: 3
  png:IHDR.color_type: 3 (Indexed)
  png:IHDR.interlace_method: 0 (Not interlaced)
  png:IHDR.width,height: 1280, 853
  png:PLTE.number_colors: 5
  png:sRGB: intent=0 (Perceptual Intent)
  png:text: 11 tEXt/zTXt/iTXt chunks were found
  png:tIME: 2017-08-10T01:55:47Z
  signature: 31db397fc7b3849a94b48bc3a7d59a2c1cfa89b78a447ab73fa044502f38d362
  Software: Le flag est : jXeQ8Bi97z956tE34weXhVHtA3
  Source: Le flag est : WVm9YT2XZg8wf5CjTee5m3f839
  Warning: Le flag est : 6w3D36WrkB4pqWRBNt2f89uf7X
Artifacts:
  verbose: true
Tainted: False
Filesize: 3559B
Number pixels: 1091840
Pixels per second: 36.3947MP
User time: 0.030u
Elapsed time: 0:01.030
Version: ImageMagick 7.0.8-14 Q16 x86_64 2018-10-25 https://imagemagick.org
```

FIGURE *2.4 – Données EXIF de l'image* `flag.png`

3. Pénultième machine virtuelle

3.1 Synthèse

TODO

3.1.1 Vulnérabilités observées

Identifiant	Criticité	Description de la vulnérabilité
V00	Important	Service d'accès à distance accessible

TABLE *3.1 – Récapitulatif des vulnérabilités*

3.1.2 Recommandations proposées

Identifiant	Vulnérabilités couvertes	Description de la recommandation	Priorité de traitement	Difficulté de mise en œuvre
R00	V00	Filtrer les flux n'étant pas nécessaires au fonctionnement de l'application	Haute	Faible

TABLE *3.2 – Récapitulatif des recommandations*

3.2 Tests d'intrusion

3.2.1 Reconnaissance active

3.2.1.1 Identification du système d'exploitation

```
~$ nmap -sV -A 192.168.122.54
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-06 12:10 CET
Nmap scan report for 192.168.122.54
Host is up (0.00050s latency).
Not shown: 566 closed ports, 430 filtered ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1.2 (protocol 2.0)
| ssh-hostkey:
|   1024 9b:ad:4f:f2:1e:c5:f2:39:14:b9:d3:a0:0b:e8:41:71 (DSA)
|_  2048 85:40:c6:d5:41:26:05:34:ad:f8:6e:f2:a7:6b:4f:0e (RSA)
80/tcp  open  http        Apache httpd 2.2.8 ((Ubuntu) PHP/5.2.4-2ubuntu5.6 with Suhosin-Patch)
|_http-server-header: Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.6 with Suhosin-Patch
|_http-title: Site doesn't have a title (text/html).
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.0.28a (workgroup: WORKGROUP)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: 2h29m58s, deviation: 3h32m07s, median: -1s
|_nbstat: NetBIOS name: ENSIIE 3, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb-os-discovery:
|   OS: Unix (Samba 3.0.28a)
|   NetBIOS computer name:
|   Workgroup: WORKGROUP\x00
|_  System time: 2018-12-06T06:10:35-05:00
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
|_smb2-time: Protocol negotiation failed (SMB2)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 29.02 seconds
```

FIGURE *3.1 – Identification de système d'exploitation et scan de ports ouverts sur le système cible*

3.2.1.2 Identification des services

TODO

3.2.2 Exploitation

TODO

3.3 Obtention du drapeau

L'accès super-utilisateur nous a directement permis de trouver un fichier `flag.txt` dans son dossier personnel. Le drapeau de cette machine est enfin **va8j2mK5f7ZeSb8Z6U5t44KCqL**.

4. Quatrième machine virtuelle

4.1 Synthèse

TODO

4.1.1 Vulnérabilités observées

Identifiant	Criticité	Description de la vulnérabilité
V00	Important	Service d'accès à distance accessible

TABLE *4.1 – Récapitulatif des vulnérabilités*

4.1.2 Recommandations proposées

Identifiant	Vulnérabilités couvertes	Description de la recommandation	Priorité de traitement	Difficulté de mise en œuvre
R00	V00	Filtrer les flux n'étant pas nécessaires au fonctionnement de l'application	Haute	Faible

TABLE *4.2 – Récapitulatif des recommandations*

4.2 Tests d'intrusion

4.2.1 Reconnaissance active

4.2.1.1 Identification du système d'exploitation

```
~$ nmap -sV -A 192.168.122.165
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-06 12:10 CET
Nmap scan report for 192.168.122.165
Host is up (0.00042s latency).
Not shown: 998 filtered ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 5.9p1 Debian 5ubuntu1.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 66:8c:c0:f2:85:7c:6c:c0:f6:ab:7d:48:04:81:c2:d4 (DSA)
|   2048 ba:86:f5:ee:cc:83:df:a6:3f:fd:c1:34:bb:7e:62:ab (RSA)
|_  256 a1:6c:fa:18:da:57:1d:33:2c:52:e4:ec:97:e2:9e:af (ECDSA)
80/tcp open  http    lighttpd 1.4.28
|_http-server-header: lighttpd/1.4.28
|_http-title: Site doesn't have a title (text/html).
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.54 seconds
```

FIGURE *4.1 – Identification de système d'exploitation et scan de ports ouverts sur le système cible*

4.2.1.2 Identification des services

TODO

4.2.2 Exploitation

TODO

4.3 Obtention du drapeau

Le niveau de privilège root nous permet de lister le contenu de son dossier personnel et de trouver le fichier `flag.txt`. Le drapeau obtenu est finalement : **C34H9krcNwp987KbUw9vuNS98L**.