|  |
| --- |
| ESEO |
| Rapport de Pentest |
| Internal Pentest |

|  |
| --- |
| EL MONCER Akil  21/11/2024 |

# Mission

## Objectifs

L’objectif principal de la mission est d’identifier les vulnérabilités potentielles et de renforcer la sécurité de l’environnement informatique de VULN.LAB. Les objectifs spécifiques incluent :

* Identifier les failles de sécurité.
* Simuler des attaques pour tester la résilience du système.
* Fournir des recommandations pratiques.
* Accéder au fichier students\_to\_contact.txt sur le partage réseau CurriculumVitae de la machine DC01 et y ajouter son Nom.

## Périmètre

La mission couvre les machines suivantes :

* 192.168.40.36 (LINX01)
* 192.168.40.31 (SRV01)
* 192.168.40.127 (DC01)

## Méthodologie

La méthodologie employée comprend :

* Connexion au VPN via un fichier OVPN.
* Scan réseau et découverte des services exposés.
* Tests d’injections SQL pour énumération des bases de données.
* Brute force et cracking des mots de passe.
* Tests de privilèges sur Active Directory avec BloodHound.
* Exploitation des partages SMB et modification du fichier cible.

# Résumé

Lors de ce test d'intrusion, plusieurs vulnérabilités critiques ont été identifiées, compromettant gravement la sécurité de l'infrastructure de l'organisation VULN.LAB :

1. Accès initial :
   * L’accès initial a été obtenu en utilisant une combinaison de techniques, notamment l’exploitation d’une injection SQL, qui a permis d’extraire des informations sensibles de la base de données, notamment les mots de passe des utilisateurs.
   * Le mot de passe d’Alice, réutilisé pour l’Active Directory, a permis de progresser dans l’attaque.
2. Découverte et cartographie réseau :
   * À l'aide de scans réseau et d'outils tels que Nmap, les services exposés des machines cibles ont été identifiés.
   * Les partages SMB et les configurations non sécurisées ont été exploités pour accéder à des fichiers sensibles.
3. Exploitation des vulnérabilités :
   * Une vulnérabilité IDOR (Insecure Direct Object Reference) a permis d’accéder aux profils d'autres utilisateurs simplement en modifiant un paramètre dans l'URL.
   * Une attaque brute force a été menée sur la page de connexion, exploitant des mots de passe faibles pour accéder à des comptes utilisateurs.
   * Une injection PHP dans les métadonnées d’une image a révélé des chemins sensibles et des fichiers critiques sur le serveur.
4. Active Directory et escalade de privilèges :
   * Les permissions insuffisamment restreintes d’Alice ont permis de lancer une analyse complète de la hiérarchie Active Directory avec BloodHound.
   * Une attaque Kerberoasting a permis de récupérer et de casser des tickets Kerberos pour des comptes de service, ouvrant la voie à une élévation des privilèges.
   * Le compte machine akil a été créé pour contourner des restrictions et obtenir des droits d'administration sur le réseau.
5. Accès final et exfiltration :
   * Le compte akil, avec des privilèges élevés, a permis d’accéder au dossier CurriculumVitae et de modifier des fichiers sensibles, comme le fichier students\_to\_contact.txt.
   * Les outils comme Lsassy ont été utilisés pour extraire des informations sensibles directement depuis le serveur cible.
6. Impact global :
   * Les vulnérabilités identifiées permettent de compromettre intégralement le réseau, avec des risques significatifs d'espionnage, de sabotage, et d'accès à des données sensibles.
   * L’infrastructure actuelle présente des faiblesses critiques liées à la gestion des mots de passe, au contrôle des accès, et au cloisonnement des privilèges.

# Synthèses des vulnérabilités

Le tableau ci-dessous présente les vulnérabilités qui ont été identifiées lors du test d’intrusion.

|  |  |  |
| --- | --- | --- |
| **Référence** | **Nom de la vulnérabilité** | **Sévérité** |
| 5.1 | Attaque brute force | Élevée |
| 5.2 | Injection PHP dans une image | Élevée |
| 5.3 | Accès non restreint à info.php | Élevée |
| 5.4 | IDOR | Critique |
| 5.5 | Injection SQL | Critique |
| 5.6 | Réutilisation de mot de passe | Critique |
| 5.7 | Accès à la hiérarchie de l’AD avec Bloodhound | Élevée |
| 5.8 | Active Directory | Critique |

# Scan réseau

## Scan de 192.168.40.36 (LINX01)

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

## Scan de 192.168.40.127 (DC01)

Une image contenant texte, capture d’écran, Police, menu

Description générée automatiquement

## Scan de 192.168.40.31 (SRV01)

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

# Fiches de vulnérabilités

Cette partie présente les différentes vulnérabilités identifiées durant le test d’intrusion.

## Attaque Brute force

#### Sévérité – Élevé

**CVSS Score:** 8.6

**CVSS Vector:** [AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:N/A:N](https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:N/A:N&version=3.1)

**Description**

L'attaque brute force consiste à tester systématiquement des combinaisons de noms d'utilisateur et de mots de passe afin de découvrir des informations d'identification valides. À l'aide de l'outil Burp Suite, nous avons pu simuler une attaque brute force sur la page de connexion disponible à l'adresse suivante :

<http://internal.vuln.lab/index.php>

Accès au formulaire de connexion :  
Nous avons saisi un nom d'utilisateur et un mot de passe aléatoires (par exemple : "azerty/azerty") pour capturer la requête dans Burp Suite.

Une image contenant capture d’écran, texte, logiciel, Logiciel multimédia

Description générée automatiquement

**Interception de la requête :**  
La tentative de connexion a été interceptée dans Burp Suite, où nous avons visualisé la requête POST envoyée au serveur contenant les paramètres username et password.

Une image contenant texte, capture d’écran, logiciel, nombre

Description générée automatiquement

Nous l’envoyons ensuite dans l’intruder :

Une image contenant texte, capture d’écran, nombre, logiciel

Description générée automatiquementUne image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

**Configuration de l'attaque dans l'Intruder :**  
Nous avons envoyé cette requête dans l'outil Intruder de Burp Suite pour configurer une attaque brute force en mode "Cluster Bomb Attack".

Une image contenant texte, Police, ligne, nombre

Description générée automatiquement

**Payload 1 :** Liste des noms d'utilisateur **Payload 2 :** Liste des mots de passe

possibles (username.txt) possibles (password.txt).

Une image contenant texte, capture d’écran, nombre, affichage

Description générée automatiquement Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement

**Filtrage des résultats :**  
Dans les paramètres de l'attaque, nous avons défini un filtre Grep-Match sur la chaîne "KO" (présente dans les réponses en cas d'échec). Cela permet d'identifier les combinaisons réussies, où "KO" n'est pas visible.

Une image contenant texte, capture d’écran, affichage, nombre

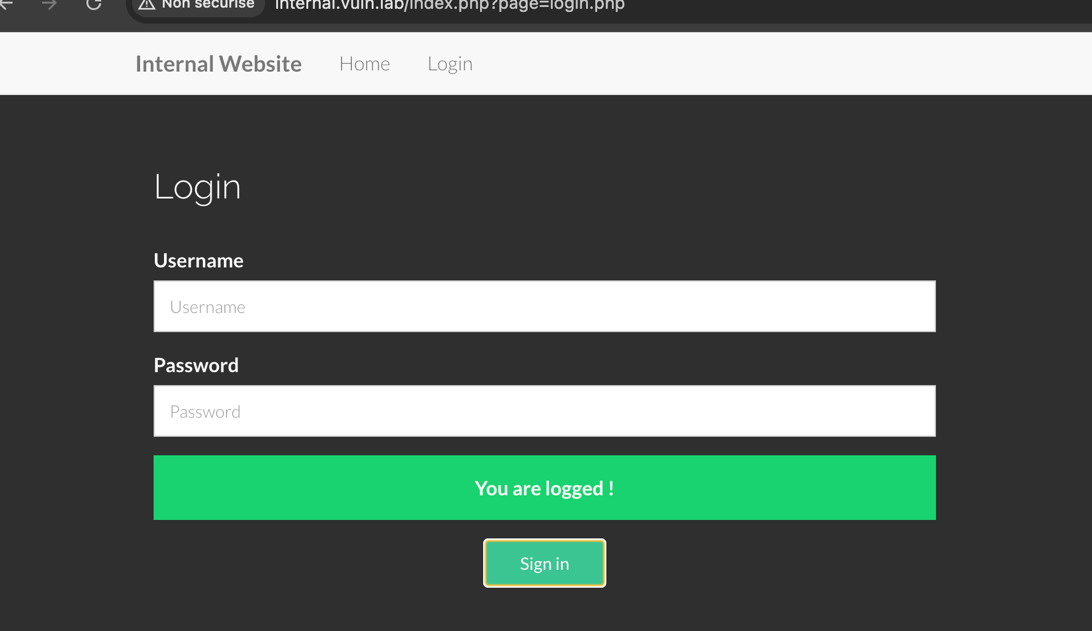
Description générée automatiquement

**Lancement de l'attaque :**  
Une fois l'attaque lancée, Intruder a testé toutes les combinaisons. La seule combinaison où "KO" n'était pas présent correspondait aux identifiants valides que je ne pourrais pas communiquer pour des raisons de sécurité.

Une image contenant texte, ligne, capture d’écran, reçu

Description générée automatiquement



**Connexion réussie :**  
En utilisant ces identifiants, nous avons pu accéder au compte.

Une image contenant texte, capture d’écran

Description générée automatiquement

Une image contenant texte, capture d’écran

Description générée automatiquement

#### Impact

#### Cette vulnérabilité expose les comptes utilisateur à des compromissions massives en cas d'attaque brute force réussie.

#### Les attaquants peuvent accéder à des données sensibles ou compromettre l'ensemble du système en obtenant un compte à privilèges élevés.

#### Correction

#### Implémentation de politiques de verrouillage : Bloquer les comptes après plusieurs tentatives de connexion échouées.

#### Mécanismes de détection : Mettre en place des outils de détection des comportements suspects comme des tentatives de brute force.

#### Renforcement des mots de passe : Exiger des mots de passe forts avec des critères minimaux (longueur, complexité, etc.).

#### Référence

<https://portswigger.net/burp>

## Injection PHP dans une image

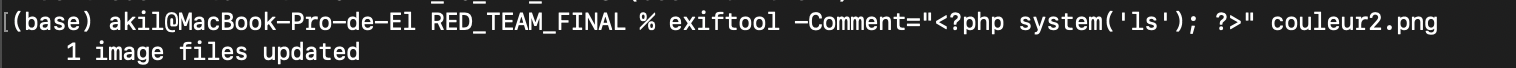
#### Sévérité – Élevée

**CVSS Score: 5.8**

**CVSS Vector:** [AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:N/A:N](https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:N/A:N&version=3.1)

**Description**:

À l'aide de la commande exiftool, nous avons injecté une ligne de code PHP dans les métadonnées d'une image afin d'exploiter une vulnérabilité dans le système de gestion des fichiers du site.  
Voici la commande utilisée :



L'image modifiée, couleur2.png, a ensuite été téléchargée via la page de test disponible à cette adresse :  
<http://internal.vuln.lab/test/upload.php>

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Une fois l'image téléchargée, en visitant le fichier à l'adresse suivante :  
<http://internal.vuln.lab/index.php?page=../test/uploads/couleur2.png>,  
le code PHP inséré dans l'image est exécuté sur le serveur. Cela nous permet d'obtenir une liste des fichiers présents dans le répertoire cible, révélant des chemins sensibles tels que ajax.php, config.php, info.php, ou encore des répertoires comme pages

Une image contenant texte, capture d’écran

Description générée automatiquement **Impact**

* + Un attaquant peut exécuter du code arbitraire sur le serveur, compromettant ainsi l'intégrité et la confidentialité des données.
  + Cette faille peut également servir de pivot pour lancer des attaques plus avancées, comme une prise de contrôle complète du serveur.

#### Correction

Validation des fichiers uploadés : Mettre en place une vérification stricte des types de fichiers autorisés (par exemple, uniquement .jpg, .png sans métadonnées exécutables).

#### Référence

https://exiftool.org/

## Accès non restreint à info.php

#### Sévérité – Élevée

**CVSS Score:** 5.8

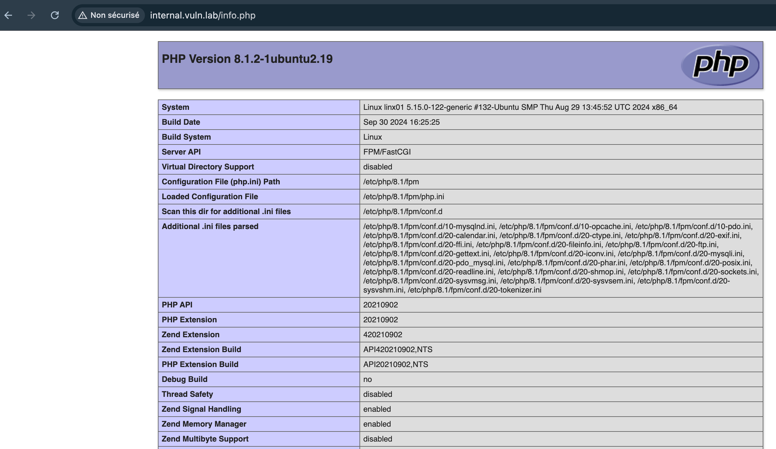
**CVSS Vector:** [AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:N/A:N](https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:N/A:N&version=3.1)

**Description**

Une vulnérabilité critique est identifiée sur la page info.php. En accédant simplement à l'URL suivante :  
http://internal.vuln.lab/info.php, il est possible d'obtenir des informations sensibles sur le serveur. Ces informations incluent :

* La version exacte de PHP utilisée (8.1.2-1ubuntu2.19).
* Le système d'exploitation (Linux 5.15.0).
* Les extensions PHP activées.
* Les chemins de configuration.
* D'autres détails qui pourraient permettre à un attaquant de préparer des attaques spécifiques.

Aucune authentification ou privilège n'est requis pour accéder à cette page. Cela rend l'information exposée accessible à toute personne ayant connaissance de l'URL.

****

**Impact**

* + Fuite d'informations : Les détails sensibles du serveur permettent à un attaquant de préparer des attaques ciblées (exemple : exécution de code à distance, injection SQL, etc.).
  + Facilitation des attaques : Ces informations peuvent aider à exploiter d'autres vulnérabilités ou permettre une escalade des privilèges.

#### Correction

Restreindre l'accès à la page info.php:

* + Supprimez cette page du serveur si elle n'est pas nécessaire.
  + Si elle est nécessaire pour l'administration, configurez des restrictions d'accès via un contrôle d'accès basé sur l'IP ou une authentification.

## IDOR (Insecure Direct Object Reference)

#### Sévérité – Critique

**CVSS Score:** 8.6

**CVSS Vector:** [AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:N/A:N](https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:N/A:N&version=3.1)

**Description**

Une vulnérabilité de type IDOR (Insecure Direct Object Reference) permet à un utilisateur non autorisé d'accéder aux données d'autres utilisateurs en modifiant simplement l'identifiant (ID) dans l'URL.  
Dans cet exemple, en accédant à l'URL suivante :  
<http://internal.vuln.lab/index.php?page=profil.php&id=5> , l'utilisateur visualise le profil de test. Cependant, si l'attaquant modifie la valeur de l'ID, par exemple en remplaçant id=5 par id=1, il peut consulter le profil d'un autre utilisateur sans restriction ni vérification des autorisations.

Ici, on peut voir le profil d’alice (ID = 5) : Et ici le profil de John (ID = 1) :

Une image contenant texte, logiciel, Logiciel multimédia, Page web

Description générée automatiquementUne image contenant texte, capture d’écran

Description générée automatiquement

**Impact**

#### Cette vulnérabilité expose les données sensibles des utilisateurs de l'application. Un attaquant peut accéder aux informations personnelles telles que les adresses mail et aux informations sensibles comme le statut et les salaires.

#### Correction

Remplacez les identifiants sensibles directement dans l’URL (comme id=5) par des identifiants indirects ou des clés chiffrées.

#### Référence

<https://www.vaadata.com/blog/fr/failles-idor-principes-attaques-exploitations-mesures-tests-securite/>

## Injection SQL

#### Sévérité – Critique

**CVSS Score:** 9.1

**CVSS Vector:** [AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:L/A:N](https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:L/A:N&version=3.1)

#### Description

Pour détecter et exploiter une vulnérabilité d'injection SQL, nous avons utilisé sqlmap à l'aide de la commande suivante :  




Le cookie requis a été extrait grâce à Burp Suite, en interceptant la requête POST envoyée lors de l'authentification. Voici la requête capturée Une image contenant texte, capture d’écran, Police

Description générée automatiquement



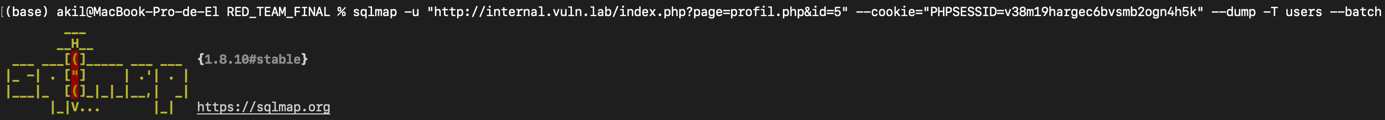
Une fois la commande exécutée, sqlmap a révélé l’existence d’une table nommée users. Cette table contient des informations critiques.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

**Exécution de la commande pour extraire les données**

Ensuite, nous avons extrait le contenu complet de la table users à l'aide de la commande suivante :





Une image contenant texte, capture d’écran, Police

Description générée automatiquement



Voici les résultats obtenus :

#### Nous avons récupéré des informations sensibles :

#### Identifiants utilisateur (username, mail)

#### Mots de passe hashés

#### Salaires

#### Descriptions des rôles

#### Par exemple, le mot de passe d’Alice a pu être récupéré en clair.

#### Impact

#### L’exploitation de cette vulnérabilité d'injection SQL permettrait à un attaquant de :

#### Récupérer des informations sensibles des utilisateurs de l’entreprise (adresses e-mail, mots de passe, salaires, etc.).

#### Escalader les privilèges en récupérant des identifiants d’administrateurs stockés dans la base.

#### Altérer ou supprimer des données critiques dans le système.

#### Correction

Renforcer les mots de passe, notamment celui d'Alice, et chiffrer les données sensibles (salaires, identifiants, rôles, etc.).

#### Référence

<https://www.vaadata.com/blog/fr/sqlmap-loutil-pour-identifier-et-exploiter-des-injections-sql/>

## Réutilisation de mot de passe

#### Sévérité – Critique

**CVSS Score:** 9.6

**CVSS Vector:** [AV:N/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:L](https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:L&version=3.1)

**Description**

Lors de notre tentative d'accès à l'Active Directory (AD), nous avons testé si Alice, dont le mot de passe pour le site web interne était connu, utilisait ce même mot de passe pour se connecter à son ordinateur.

En testant cet identifiant avec la commande suivante, nous avons pu confirmer que Alice réutilise le même mot de passe pour plusieurs services, notamment son compte Active Directory. Cela nous a permis d’accéder à différents partages réseau avec les droits associés à son compte.

Une image contenant texte, capture d’écran, Logiciel multimédia, Police

Description générée automatiquement



#### Impact

L'impact de cette vulnérabilité est significatif :

1. Accès non autorisé à des ressources sensibles : Les partages réseau accessibles via le compte d'Alice pourraient contenir des données critiques.
2. Escalade de privilèges potentielle : Si d'autres comptes d'utilisateurs ou de services sensibles réutilisent leurs mots de passe, cela pourrait conduire à un contrôle total de l'infrastructure.
3. Compromission de l'ensemble de l'environnement AD : L'utilisation d'un mot de passe faible ou réutilisé constitue une porte d'entrée majeure pour les attaquants.

#### Correction

Pour corriger cette vulnérabilité :

1. Mettre en place une politique de mots de passe solides et uniques : Utiliser des mots de passe aléatoires, complexes et différents pour chaque service.
2. Configurer une rotation des mots de passe : Instaurer des politiques qui obligent les utilisateurs à changer leurs mots de passe régulièrement.

## Accès à la hiérarchie de l’Active Directory avec BloodHound

#### Sévérité – Élevée

**CVSS Score :** 8.7

**CVSS Vector :** [AV:N/AC:L/PR:L/UI:R/S:C/C:H/I:H/A:N](https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:R/S:C/C:H/I:H/A:N&version=3.1)

**Description**

Nous avons pu, à l'aide des identifiants d'Alice, collecter des informations essentielles sur la structure du réseau Active Directory (AD). Cette commande a permis de générer plusieurs fichiers JSON contenant des données critiques :

* Liste des administrateurs de domaine.
* Liste des utilisateurs avec des droits particuliers.
* Relation entre les groupes, utilisateurs et permissions au sein du réseau.

Ces fichiers JSON peuvent ensuite être importés dans l’outil BloodHound, permettant une visualisation complète et précise de la hiérarchie et des permissions du réseau Active Directory. Cela donne une vue d'ensemble sur :

* Qui possède des droits d'administrateur.
* Quels groupes ont des permissions critiques.
* Quels chemins d’escalade de privilèges peuvent être exploités.

Cela représente une menace majeure pour la sécurité de l’organisation, car un attaquant peut :

* Identifier les points faibles de la configuration AD.
* Escalader les privilèges en exploitant les relations hiérarchiques.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement



**Impact**

Exposition des informations critiques de l’AD :

* + Permet de comprendre comment les utilisateurs, groupes et machines interagissent.
  + Donne une feuille de route pour cibler des utilisateurs spécifiques ou des systèmes avec des privilèges élevés.

#### Correction

* + Activer l’audit des actions LDAP :

Configurer des journaux d'audit pour détecter les requêtes suspectes via des outils comme BloodHound.

* + Implémenter l’authentification multifactorielle :

Ajouter un second facteur d'authentification pour accéder aux ressources sensibles, même sur des comptes standards.

**Reference**

https://www.it-connect.fr/chapitres/bloodhound-installation-avec-docker/

## Active Directory

#### Sévérité – Critique

**CVSS Score :** 9,8

**CVSS Vector :** [AV:N/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:H](https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:H&version=3.1)

**Description :**

Pour pénétrer l’Active Directory (AD), nous avons utilisé le mot de passe d’Alice récupéré par une injection SQL. Ce mot de passe a permis un accès initial à ses droits Active Directory. En exploitant les vulnérabilités suivantes, nous avons compromis les droits de plusieurs comptes et accédé aux ressources sensibles du réseau.

**1. Kerberoasting :**

L’attaque a permis de récupérer les tickets Kerberos pour les comptes de service.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement



Le hash du compte svc\_srv a été extrait pour une attaque par force brute.

**2. Crackage du hash avec Hashcat :**

Hashcat a permis de récupérer le mot de passe de svc\_srv :

Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement



**3. Création d’un compte machine :**

Pour contourner les restrictions d’Alice, un compte machine nommé akil a été créé avec la commande BloodyAD :





**4. Utilisation de BloodHound :**

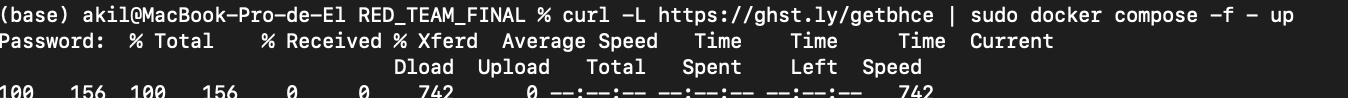
L’analyse des droits et permissions Active Directory a été effectuée via BloodHound en deux étapes :

* + Génération des fichiers JSON contenant la topologie des droits dans l’AD
  + Installation et démarrage de l’interface BloodHound

Une image contenant texte, capture d’écran, Police

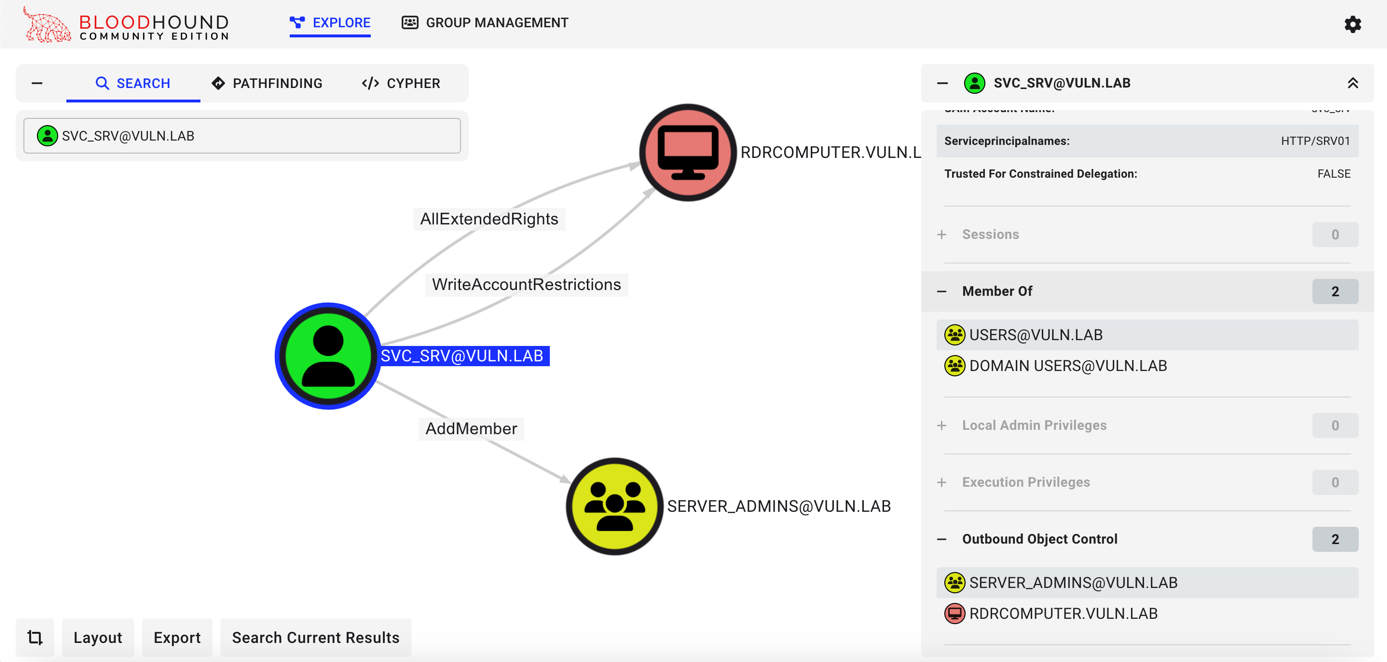
Description générée automatiquement





**5. Élévation des privilèges avec svc\_srv :**

En utilisant les droits du compte svc\_srv, le compte akil a été ajouté au groupe server\_admins, accordant des droits d’administration sur les serveurs :







Nous vérifions ensuite qu’akil a bien les permissions nécessaires pour accéder aux informations d'un compte administrateur :

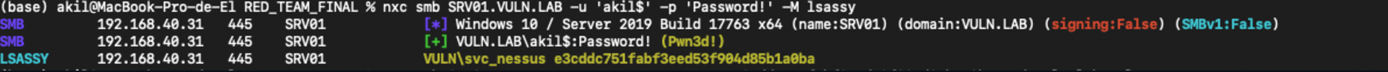
Une image contenant texte, capture d’écran, Police, Logiciel multimédia

Description générée automatiquement



**6. Extraction d’informations sensibles avec Lsassy :**

L’outil Lsassy a permis d’extraire des informations sensibles depuis le serveur SRV01 :





Nous pouvons voir un hash correspondant au compte svc\_nessus.

**7. Manipulation des groupes avec svc\_nessus :**

Avec les privilèges obtenus grâce au hash de svc\_nessus, akil a été ajouté au groupe human\_resources :

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement





**8. Accès final au dossier CurriculumVitae :**

Avec les droits du groupe human\_resources, le dossier CurriculumVitae a été accessible :

Une image contenant texte, capture d’écran, Police, Logiciel multimédia

Description générée automatiquement



**9. Modification de fichiers sensibles :**

Le fichier students\_to\_contact.txt a été téléchargé, modifié pour ajouter des informations, et renvoyé vers le serveur :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement



**Impact**

* + Contrôle complet de l’Active Directory :

Accès aux privilèges administratifs des serveurs.

Lecture, modification ou suppression de données sensibles.

Réduction significative de la sécurité du réseau.

* + Atteinte à la confidentialité des utilisateurs :

Exposition des informations personnelles et professionnelles.

* + Risque de sabotage interne :

Possibilité d’ajouter ou supprimer des membres dans des groupes critiques.

#### Correction

* + Politique stricte de mots de passe :

Imposer des mots de passe uniques, robustes et régulièrement renouvelés.

Désactiver les comptes inutilisés ou obsolètes.

* + Restreindre l’accès :

Limiter les privilèges des comptes de service et des utilisateurs standards.

Imposer des permissions strictes sur les partages SMB

#### Références

<https://specopssoft.com/blog/active-directory-attack-paths/>

<http://127.0.0.1:8080/ui/explore>

<https://hashcat.net/wiki/doku.php?id=hashcat_utils>