

Ben Janis
Denis Bravenec
Kieran Green
Maretta Morovitz
CTF Write-Up
4 March 2017 (Extended Due Date)

Executive Summary

Throughout this CTF exercise, we were able to compromise the system through simple, often non-technically advanced, means. Such straightforward, yet effective, methods underscore the serious challenge we face to build and protect secure systems. Using a Linux distribution such as Kali, even a non-technical individual has all of the necessary tools to successfully disrupt an advanced technical system. These findings led us to a number of policy recommendations including: to patch systems regularly, to limit attack surface and, when this surface must be expanded, to exercise additional measures. However, as was shown through the relative simplicity of this exploiting this machine, security breaches and vulnerabilities are inevitable. Thus we present a recommendation to facilitate responsible disclosure.

Introduction

The CTF used a variety of tools, built into the Kali Linux distro for attacks. These included metasploit, nmap, and basic Google searches among others. Additional attacks against the router also relied on hardware vulnerabilities. While we were not ultimately successful in breaking into the machine before Ming provided the password hint, we had been able to gather significant evidence about the machine as well as several possible vulnerabilities. After being given the vulnerable VM, we continued to try to attack the

machine even after the CTF was over and discovered a large number of additional vulnerabilities that will be discussed in the sections to follow. It is interesting to note that all of these attacks did not require technical knowledge and could have been carried out by any “script kitty”.

Tools and methods

The team used a variety of tools and methods against both the computer and the router. Our method can be broken down into two distinct phases, pre-attack and attack. The pre-attack phase consisted of two subphases, namely reconnaissance and scanning. During the reconnaissance subphase we attempted to ascertain additional information about the device. We examined all documentation on the CTF and examined the router to determine make, model, brand, etc. In the second pre-attack phase, scanning, we started by pinging the ip to determine that the machine was live. Once we had determined that the device was indeed live, we performed an nmap scan. From this scan we found a list of open ports. By setting additional flags we found additional information pertaining to the machine including OS. We did additionally research through Google to explore vulnerabilities and known exploits.

The next phase, the attack phase, was carried out against the server, mainly using metasploit. We went through the list of open ports to build and deploy metasploit exploits that may provide us a backdoor into the machine. Additionally we used the information found on Google to find additional metasploit exploits.

We had a similar methodology for approaching the router. During our pre-attack phase, we snuck to the front of the classroom when the professors were not looking to take a picture of the underside of the router to get product info as seen in Figure 3.1.



Fig 3.1: information found on the router

We then used the product info to Google default passwords, known vulnerabilities, etc. During the attack phase, we carried out our attack first by attempting to brute force the password with default and known weak passwords. However, we did not have any success. Our next attack on the router was physical. One team members distracted Professor Taliafaro while a second member snuck to the front of the classroom. Using an iPhone reset

pin we attempted to press the router reset button, knowing that if we were able to press the button for 10 seconds we would reset the router to factory settings and be able to use the default passwords. However, unfortunately we must have not pressed the button for the full 30 seconds before other students became suspicious and the router did not fully reset. After this failed, we attempted to DDOS the router using curl. The router was indeed brought down, but it remains unclear if this was due to our attack, a classmate's, or a combination of these attacks.

Findings

A simple Nmap scan of the server quickly identifies the target's operating system as Microsoft Windows Server 2008. The scan also lists the many open ports of the server, shown in figure 4.1. Of these, the most interesting are the occasionally open 21 (ftp), 225 (remote desktop), 3000 (http server), and 3389 (a known vulnerable port).

The three users on the system can be found through compromising the admin account (username: Administrator, password: p@ssw0rd). The other two are sam and student. While we did not determine their passwords, the passwords can be reset by the admin, allowing us to access either.

There are many different programs installed on this machine, some with serious vulnerabilities. Figure 4.2 shows the software installed to Program Files, as well as a series of programs left in a directory on the Desktop called Tools. The server has installed Mozilla Firefox, Google Chrome, and Windows Internet Explorer as internet browsers. Beautifully, both Chrome and Internet Explorer each display security warnings on startup, hinting at

the probability of vulnerability to exploits. There are also two separate vulnerable servers on the system: Hacme Casino and WebGoat, both designed as practice targets for exploitation. The Tools directory contains two overtly malicious tools and one that is often malicious: havij, a SQL injection tool; HttpDosTool, a program build to attempt denial of service attacks on remote servers; and ca_setup.exe, the installer for Cain & Abel, a password recovery tool for Windows. Other tools include the HashCalc hash calculators, EasyFtp Server, the Wireshark and WinPcap network monitors, the Nmap port scanner, the TrueCrypt encryption software, Windows Mail, Java, the 7-zip compression software, the filealylz malware detector, and erase.exe and hello.exe, two trolling programs (see figure 4.4).

```

root@kali:~# nmap -A 192.168.40.128

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2017-03-03 22:01 EST
Nmap scan report for 192.168.40.128
Host is up (0.00044s latency).
Not shown: 986 closed ports
PORT      STATE SERVICE        VERSION
21/tcp    open  ftp            EasyFTP Server ftpd
|_ftp-anon: ERROR: Script execution failed (use -d to debug)
|_ftp-bounce: no banner
135/tcp   open  msrpc          Microsoft Windows RPC
139/tcp   open  netbios-ssn    Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds   Windows Server (R) 2008 Standard 6001 Service Pack
1 microsoft-ds (workgroup: WORKGROUP)
1025/tcp  open  msrpc          Microsoft Windows RPC
1026/tcp  open  msrpc          Microsoft Windows RPC
1027/tcp  open  msrpc          Microsoft Windows RPC
1028/tcp  open  msrpc          Microsoft Windows RPC
1029/tcp  open  msrpc          Microsoft Windows RPC
1030/tcp  open  msrpc          Microsoft Windows RPC
3000/tcp  open  http           WEBrick httpd 1.3.1 (Ruby 1.8.2 (2004-12-25))
|_http-server-header: WEBrick/1.3.1 (Ruby/1.8.2/2004-12-25)
|_http-title: Hacme Casino
3389/tcp  open  ms-wbt-server  Microsoft Terminal Service
|_ssl-cert: Subject: commonName=WIN-JWBPPZSXEfv
|_Not valid before: 2017-02-13T02:48:47
|_Not valid after: 2017-08-15T02:48:47
|_ssl-date: 2017-03-04T22:28:39+00:00; +19h26m16s from scanner time.
5357/tcp  open  http           Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Service Unavailable
8080/tcp  open  http           EasyFTP Server httpd
|_http-server-header: Easy-Web Server/1.0

Host script results:
|_clock-skew: mean: 19s, deviation: 1s, median: 19s
|_nbstat: NetBIOS name: WIN-JWBPPZSXEfv, NetBIOS user: <unknown>, NetBIOS MAC: 00:0c:29:5b:0c:92 (VMware)
|_smb-os-discovery:
|_OS: Windows Server (R) 2008 Standard 6001 Service Pack 1 (Windows Server (R) 2008 Standard 6.0)
|_OS CPE: cpe:/o:microsoft:windows_server_2008::sp1
|_Computer name: WIN-JWBPPZSXEfv
|_NetBIOS computer name: WIN-JWBPPZSXEfv
|_Workgroup: WORKGROUP
|_System time: 2017-03-03T17:34:08-08:00
|_smb-security-mode:
|_account used: guest
|_authentication level: user
|_challenge response: supported
|_message signing: disabled (dangerous, but default)
|_smbv2-enabled: Server supports SMBv2 protocol

```

Figure 4.1: Nmap scan results

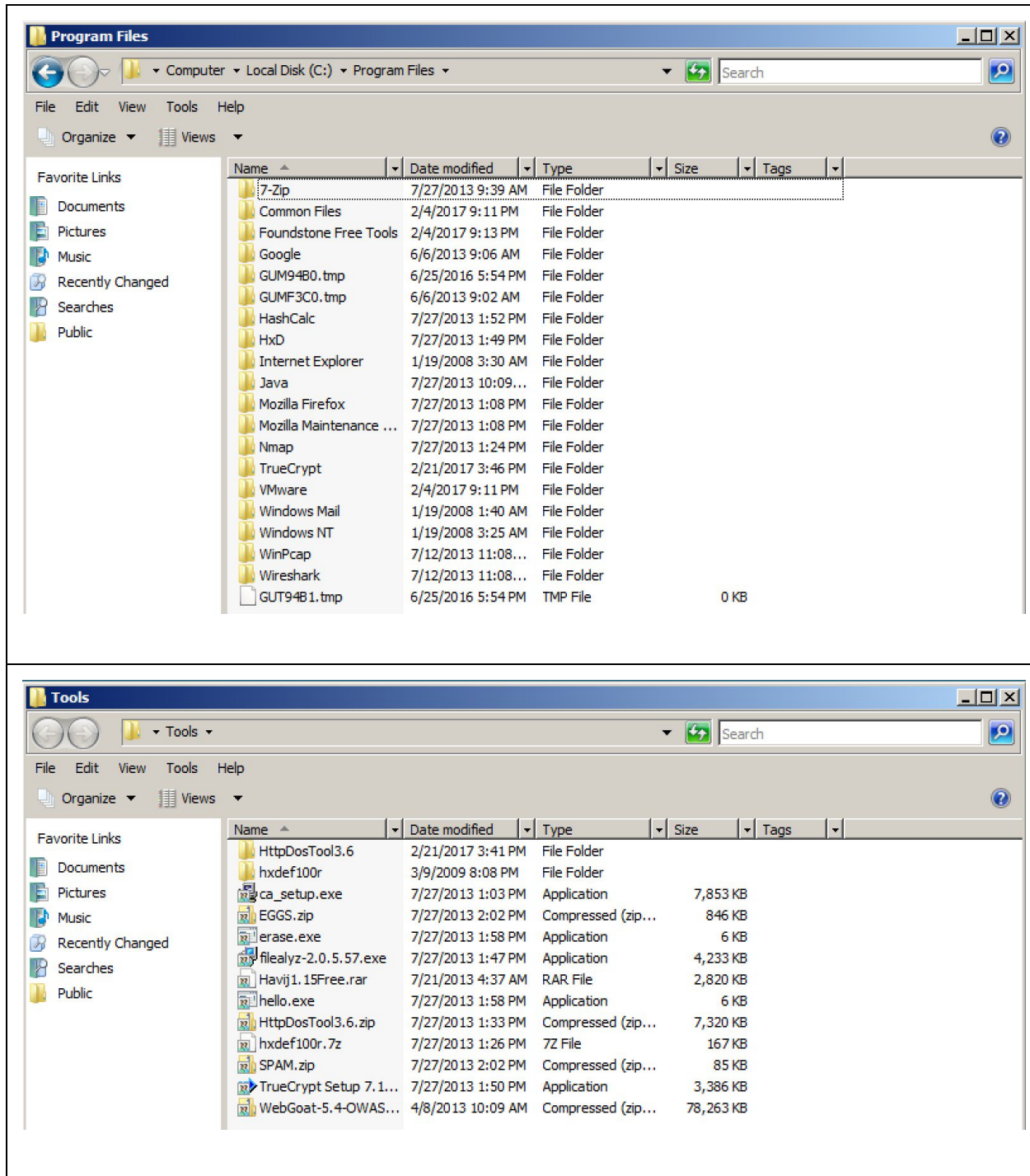
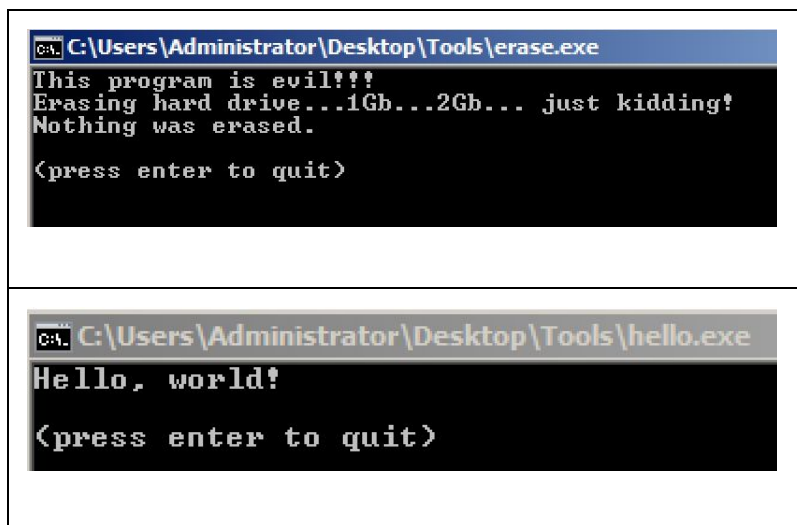


Figure 4.2: Figure 4.2a - Contents of Program Files, Figure 4.2b - Contents of Tools

This computer will no longer receive Google Chrome updates because Windows XP and Windows Vista are no longer supported.

Caution: Internet Explorer Enhanced Security Configuration is not enabled

Figure 4.3: Figure 4.3a - Google Chrome end of support message, Figure 4.3b - Internet Explorer disabled security settings message



```
C:\Users\Administrator\Desktop\Tools\erase.exe
This program is evil!!!
Erasing hard drive...1Gb...2Gb... just kidding!
Nothing was erased.

<press enter to quit>

C:\Users\Administrator\Desktop\Tools\hello.exe
Hello, world!

<press enter to quit>
```

Figure 4.4: Figure 4.4a - erase.exe, Figure 4.4b - hello.exe

By looking at the list of open ports, we were able to determine that remote desktop was enabled on the machine. Thus we were able to access the login screen via the rdesktop command as seen in Figure 4.5a. At this point we attempted to brute force the root account password. We focused on this account as gaining root access would give us escalated privileges. If able to gain root access we would have complete control over all other accounts and computer settings and data. We tried passwords such as “password”, “password123”, “admin”, blank. It was Ming’s hint that allowed us to finally arrive at the correct username/passwords combo, Administrator/p@ssw0rd. This combination resulted in successful remote login as seen in Figure 4.5b.

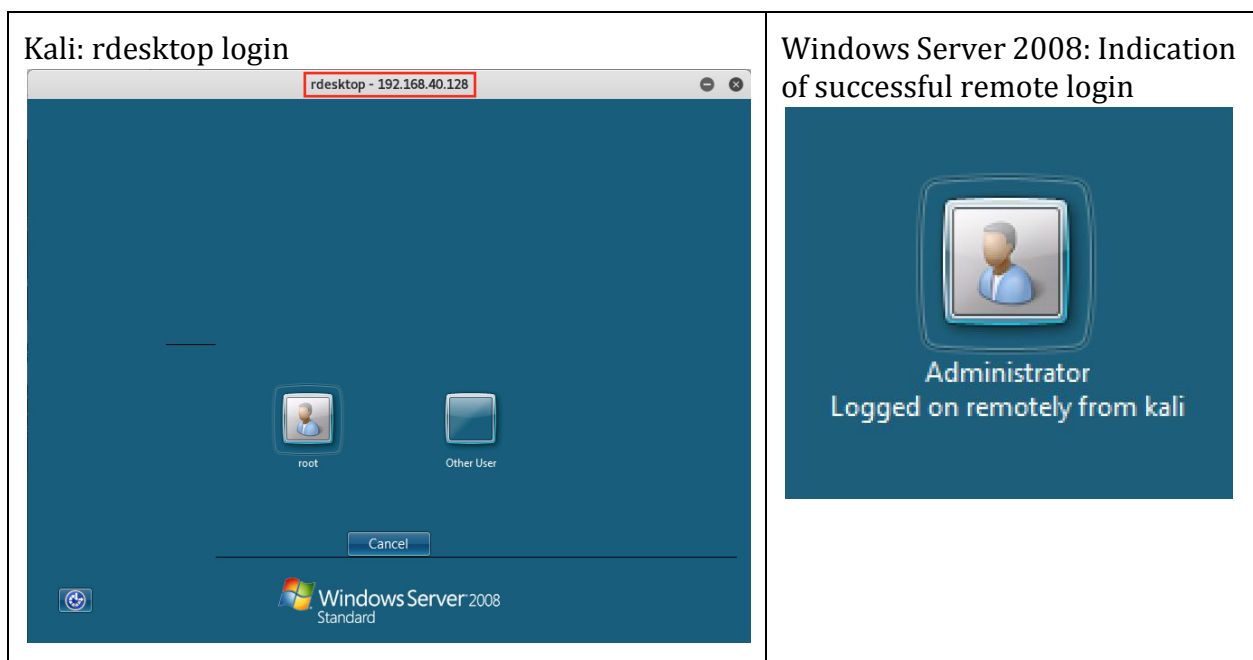


Figure 4.5: 4.5a - The rdesktop login screen, 4.5b Successful rdesktop login

In addition to providing an attack surface for brute force password attacks, rdesktop provides additional vulnerabilities for attack. Rdesktop for Windows Server 2008 is vulnerable to the “Remote Desktop Protocol Vulnerability”, according to CVE-2012-0002. This vulnerability results from rdesktop not properly processing packets in memory, thus allowing for remote attackers to execute arbitrary code by sending crafted RDP packets triggering access to an object that were either not properly initialized or have been deleted (CVE-2012-0002). Thus we were able to use a metasploit exploit to launch a DOS attack and bring down the machine. As seen in 4.6a we see the deployment of the rdesktop exploit. Found through of use auxiliary/dos/windows/rdp/ms12_020_maxchannelids, the exploit results in the screens seen in figure 4.6 b and 4.6b. The entire machine is brought down through this exploit. This violates the A in the CIA acronym, as the machine is no longer available.

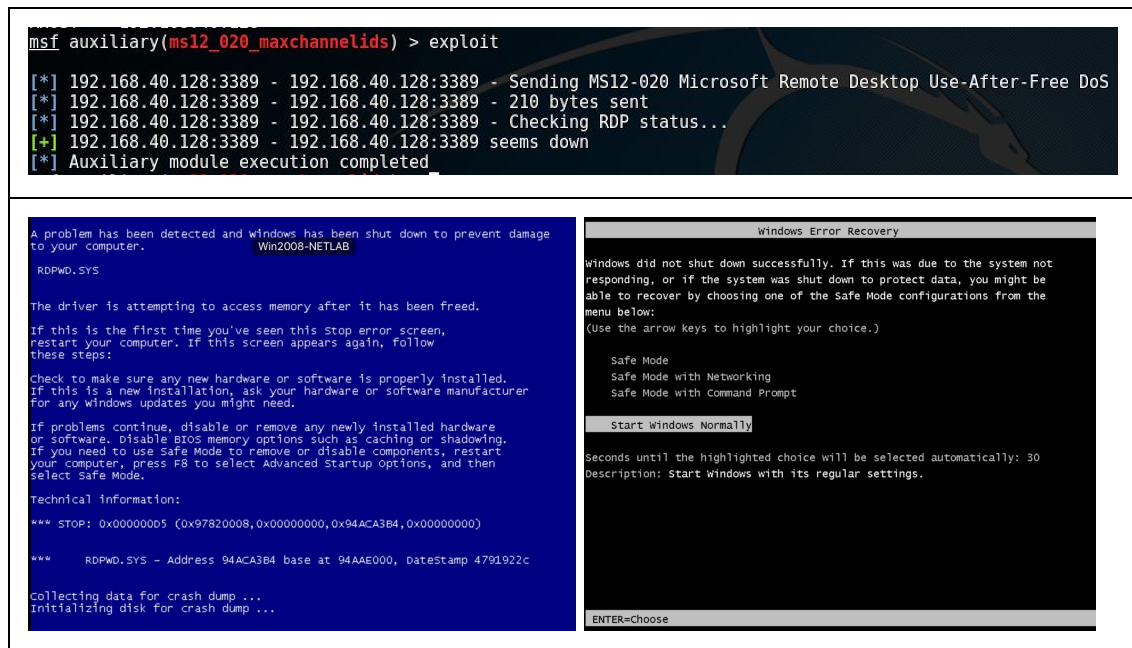
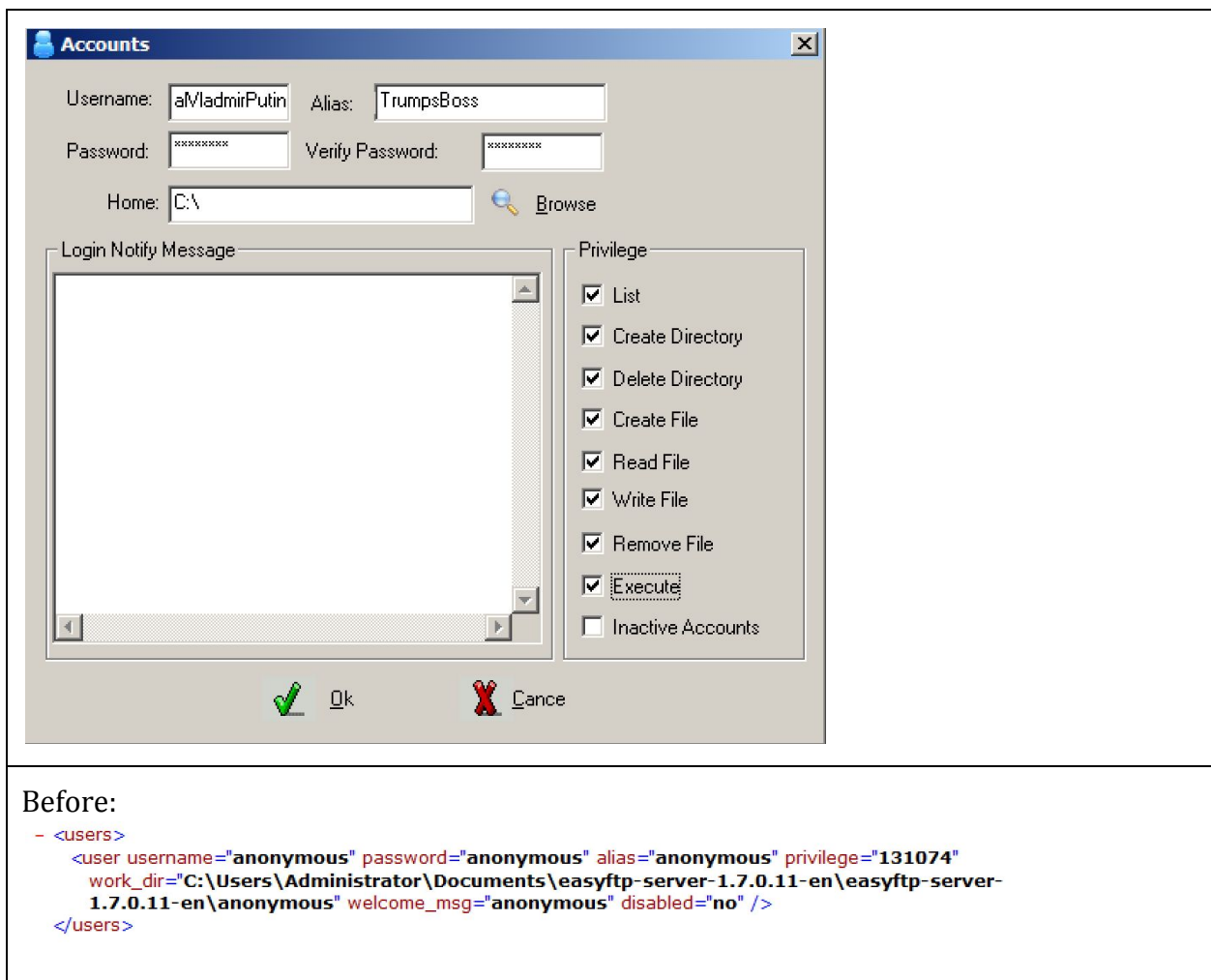


Figure 4.6: Execution and result of DOS attack (CVE-2012-0002)

From the compromised admin account, the control panel for the EasyFtp Server can be found at:

C:\Users\Administrator\Documents\easyftp-server-1.7.0.11-en\easyftp-server-1.7.0.11\Ftpconsole.exe

The control panel (shown in figure 4.7) allows for adding ftp accounts, giving us the ability to create a lightweight backdoor into the server.



After:

```
- <users>
  <user username="anonymous" password="anonymous" alias="anonymous" privilege="131074"
    work_dir="C:\Users\Administrator\Documents\easyftp-server-1.7.0.11-en\easyftp-server-1.7.0.11-en\anonymous"
    welcome_msg="anonymous" disabled="no" />
  <user username="RealVladmirPutin" password="CCC79F761B1A96A1991849ECC0C90D59" alias="TrumpsBoss"
    privilege="458783" work_dir="C:\" welcome_msg="" disabled="no" />
</users>
```

```
root@kali:~# ftp 192.168.40.128
Connected to 192.168.40.128.
220- Ftp Site Powerd by BigFoolCat Ftp Server 1.0 (meishu1981@gmail.com)
220- Welcome to my ftp server
220
Name (192.168.40.128:root): RealVladmirPutin
331 User name okay, need password.
Password:
230-
230- Ftp server have run for 0h-11m-44s
230 RealVladmirPutin logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT Command successful.
150 Opening ASCII mode data connection
drw-rw-rw-  1 user  group      0 Feb  4 21:13 lib
drw-rw-rw-  1 user  group      0 Jan 19 01:40 PerfLogs
drw-rw-rw-  1 user  group      0 Feb 21 15:46 Program Files
drw-rw-rw-  1 user  group      0 Jul 27 13:08 ProgramData
drw-rw-rw-  1 user  group      0 Jun  6 04:37 Users
drw-rw-rw-  1 user  group      0 Mar  3 17:43 Windows
-rw-rw-rw-  1 user  group     24 Sep 18 13:43 autoexec.bat
-rw-rw-rw-  1 user  group     10 Sep 18 13:43 config.sys
-rw-rw-rw-  1 user  group    403456 Jul 27 15:09 mimikatz.exe
226 Transfer complete.
ftp>
```

Figure 4.7: Figure 4.3a - Add user panel from EasyFtp Server, Figure 4.3b - ftpusers.xml before and after user addition, Figure 4.3c - Successful ftp access from Kali

One of the services available on the target machine is the Hacme Casino, a web server that offers a casino game. The website is riddled with exploits, but before we attempted to find any of them, we ended up finding the pdf on the host machine detailing every available exploit. After finding this, we focused our efforts on other facets of the machine, since we wanted to be more than script-kiddies plugging in exploits that others had found.

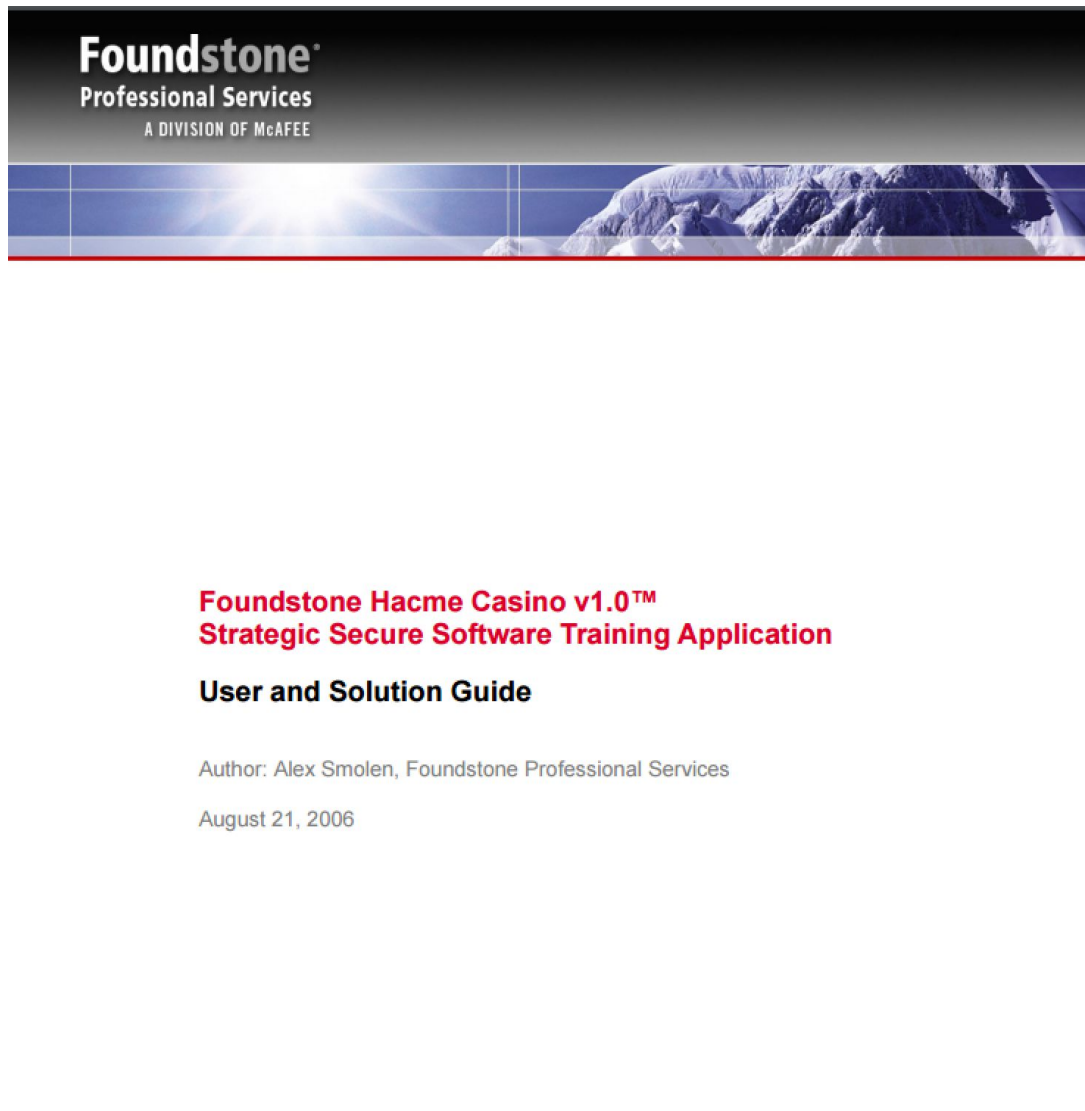


Figure 4.8: Documentation of Hacme Casino exploits

Findings as They Pertain to Policy Suggestions

The most glaring vulnerability that we encountered was the lack of a sophisticated password for the system. Although the password was not glaringly obvious (i.e., not “password,” “admin,” etc.), it was still not complex. Hence, once we acquired a clue as to what the password *might* be, breaking in was relatively simple. We concluded that the clue that Ming gave was roughly analogous to “HUMINT” about the target-- i.e., knowing behaviours or patterns that allow us to guess the password. So, the most obvious recommendation that we can give is the establishment of a uniform code of good cyber hygiene for employees in both the public and private sector, the most important of which is the maintenance of strong, complex passwords that are changed regularly.

We were also able to find and exploit number of vulnerabilities due to the target system running on software that had not yet been updated. Hence, as part of a list of “best practices” for CISOs, we recommend that all security operations centers ensure that their employees are using software with the latest available patches. Furthermore, we found that since the system’s RDesktop was open it provided another vulnerable aspect that we could access (via password guessing, etc.). We recommend that RDesktop be used sparingly, and that it should be closed when not in use to reduce the total number of vectors that an attacker could potentially use. Additionally, a strong password for RDesktop is especially important, given that it can potentially give an attacker a potentially devastating avenue of attack. When conducting our exploitation we also found that the fact that the router was open to the public internet made it particularly vulnerable. So, limiting

the contact between a firm's router and the public internet (number of ports, etc) could significantly reduce the amount of ground that SOCs have to cover.

Another unconventional vulnerability that we encountered was the physical security of the router itself. When theorizing potential vulnerabilities that we might exploit, one of the options that we considered was simply unplugging the router for 30 seconds so that the password would reset. We actually managed to achieve unplugging the router, but were unable to reset the password due to lack of an ethernet cable. Nevertheless, the fact that the router was poorly guarded rendered the system vulnerable to this kind of exploitation. Hence, our last policy recommendation is that firms should ensure that only authorized personnel have physical access to routers, and that restrictions be put in place on bringing unauthorized hardware (flashdrives, etc.) into the workplace.

In terms of policy recommendations, although it would be difficult for the federal government to mandate that each of these suggestions be put into place, we assess that it would nevertheless be valuable for the United States government to partner with private security providers to create a definitive set of "best practices" for CISOs in order to maintain security (encompassing the suggestions that we laid out in this document).

Disclosure

A significant policy implication that is based on our findings and the CTF exercise, is the question of disclosure by both the private and public sector. In summary, to what extent are organizations that handle sensitive data required to disclose vulnerabilities, breaches,

and data theft? This will serve as a guiding question for our inquiry into the intersection between policy and cybersecurity.

When an attacker breaks into a Remote Desktop system, or attempts to Brute-force to gain access, as we attempted in the exercise, the recipient of the attack is able to monitor the actions. Therefore, should all such attacks be disclosed to a government agency by law, or is it the responsibility of the individual or organization to voluntarily manage their security and disclosure of certain information? (EDIT)

The challenge with disclosing the vulnerability of a system or full disclosure of all breach attempts, is the fact that this can deal significant reputational damage to an enterprise or government institution. Additionally, by disclosing what security measures have been taken, including what latest versions of software are being used and what firewalls are in place, attackers may be enabled to find another attack vector. Avi Gesser¹, a partner at Davis Polk, an international law firm, successfully describes the complexity behind disclosure:

“Regulators like the SEC have to find the right balance between encouraging companies to be helpful with investors by accurately and fairly disclosing their risks, and helping sort out what is and what is not material for investors, while not requiring companies to provide a roadmap for hackers as to where they are

¹ Amy Terry Sheehan, "Meeting Expectations for SEC Disclosures of Cybersecurity Risks and Incidents (Part One of Two)" (The Cybersecurity Law Report. 12 Aug. 2015. Web. 2 Mar. 2017).
<https://www.davispolk.com/sites/default/files/agesser.Cybersecurity.Law_Report.aug15.pdf>.

vulnerable, or requiring disclosures that may trigger lawsuits - but that don't actually add any value."

The key terms here are *value* and *balance*. It is crucial to develop the necessary regulations that could prevent large value loss as a result of exploitation and cyber crime, especially when firms underinvest into their information security. On the other hand, poorly designed regulation can also cause value loss due to unnecessary lawsuits and the provision of roadmaps to attackers, as specified by Avi Gesser.

To address this question of balance, a recommendation would be to create a Monitoring task force, as outlined below:

- **Risk Analysis and Threat Assessment Monitoring:** the monitoring of all attacks and attempts should be fully disclosed with an inter-government agency task force center for data collection and oversight.
 - Information inflows and outflows between the private sector and the task force would be confidential.
 - Regulation will have to be designed in a way that, if there is a high risk of value loss, the government force consequent information disclosure to the public
 - The task force would serve as a preventative measure to decrease the risk of value loss due to cyberattacks

This inter-government agency task force would strike the balance between full disclosure, and nondisclosure due to the fear of public backlash and reputational damage.

Currently, a similar proactive initiative has been undertaken by the Securities and Exchange Commission (SEC), under the umbrella of its “Office of Compliance Inspections and Examinations” (OCIE). As a response to the SEC’s Cybersecurity Roundtable and examinations of cybersecurity risks, the OCIE “announced a focus on cybersecurity compliance and controls as part of its 2015 Examination Priorities... which will involve more testing to assess implementation of firm procedures and controls.” ²

While this initiative is not an active monitoring center, it is a strong step towards cybersecurity monitoring and regulation within large corporations.

Conclusion

In conclusion, although our team did not manage to gain access to the system during the time allotted, we felt that our underlying tactics, techniques, and procedures were sound. If we were to make one alteration, that would have been spending less time on trying to gain access to the router and more time trying to break into the network itself. Overall the CTF was an eyeopening experience in how easily complex systems can be brought down without significant technical work.

² Office of Compliance Inspections and Examinations, "OCIE's 2015 Cybersecurity Examination Initiative" (U.S. Securities and Exchange Commission, 15 Sept. 2015. Web. 27 Feb. 2017).
<<https://www.sec.gov/ocie/announcement/ocie-2015-cybersecurity-examination-initiative.pdf>>.

Bibliography

Office of Compliance Inspections and Examinations. "OCIE's 2015 Cybersecurity Examination Initiative." (n.d.): n. pag. U.S. Securities and Exchange Commission, 15 Sept. 2015. Web. 27 Feb. 2017.

<<https://www.sec.gov/ocie/announcement/ocie-2015-cybersecurity-examination-initiative.pdf>>.

Sheehan, Amy Terry. "Meeting Expectations for SEC Disclosures of Cybersecurity Risks and Incidents (Part One of Two)." The Cybersecurity Law Report (n.d.): n. pag. 12 Aug. 2015. Web. 2 Mar. 2017.

<https://www.davispolk.com/sites/default/files/agesser.Cybersecurity.Law_Report.aug15.pdf>.

The MITRE Corporation. "Common Vulnerabilities and Exposures." CVE - CVE-2012-0002. N.p., n.d. Web. 04 Mar. 2017.

<<http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2012-0002>>