The Linux Trainer

Dan Goldsmith

Chapter 1: The Linux Trainer

Introduction

Being comfortable with the command line is a vital skill for the ethical hacker. Remote shells (which are predominately command line based) are an important part of ethical hacking. You will be spending a lot of time using them during penetration tests, and if you take part in CTF competitions. The majority of exploits start with a shell environment. This provides the initial 'toehold' on a system that allows us to dig deeper, and strengthen our grip on the system.

This week, we are going to take a look at Basic Linux command line use. We will introduce some core concepts getting around the system using the command line, creating and editing files, and then look at some basic Enumeration concepts such as Permissions, finding things and files, and a simple (but pretty cool) privesc exploit.

> These are all useful commands that will you should be comfortable with, don't worry too much if you cant remember the commands at first, keep a notebook (see Hints and Tips section), of the useful commands for reference, they will soon become "Muscle Memory" (and if they don't, you will have a cheat sheet to remind you)

The Quest for Linux-Fu

Work through the Linux Trainer, and complete as many Levels as you can. When you complete each level you will get a password for the next.

The instructions for each level are included on the machine, you can view them using the instructions provided on the server (Hint, finding how to read the files containing the instructions is a Level in its own right)

RTFM!!!

You may notice that some levels do not provide *all* the information you need to complete the task. Welcome to the world of "Background Reading", you are strongly encouraged to do your own research on the topics we introduce, not only is it part of University life but most hacking problems wont have an answer on stack overflow.

Secure Shell (SSH)

To get access to the server we need a client to access the remote terminal. In this case we will use SSH, this is probably the most common way of getting a shell on a remote service. Unlike Telnet, which communicates in plain text, SSH allows encrypted communication to a server.

When using a terminal to connect over SSH the command consists of three parts:

```
ssh <user>@<host>
```

- **ssh** is the command name
- **user** is the username to login as
- **host** is the service you wish to connect to, which can be either an IP address or a domain name

So to SSH into the **hacker** account on **example.org** we would use:

```
ssh hacker@example.org
```

While traditionally Windows users would connect over SSH using tools such as **Putty** or **Moba Xterm** (see further reading if you're interested). More modern version of Windows (from Windows 10 18.09) include an SSH client by default in the command line. The command to access the SSH client is the same as that for Linux.

Native Windows SSH?

Yeah, I know, it blew my mind when I discovered that there was native windows SSH. Lets hope the current love affair with 'Nix based tools continues. However, Mobaterm deserves an Honorable mention, it also includes a bash shell, FTP connectivity, and methods of connecting to other remote services such as Telnet or Serial lines. Its worth checking out.

Instructions for Connecting to the trainer.

You will also need to connect to the server that holds the challenges, for this we will use the SSH Client. Unfortunately, the instructions differ depending on whether you are in the Labs (Cisco and EH [EC1-13, EC1-14]) or the rest of the University. This is due to the way the DNS resolution works in the Labs.

You can have the first set of login details for free:

- Username: *level0*
- Password: *level0*

Connecting in the Hacking Labs

The address of the server is: **zaphod.cueh**

(NOTE: On some machines the DNS server may not be working properly. In this case use the IP address of *192.168.1.20*)

Therefore the command to use SSH on the command line will be.

```
ssh level0@zaphod.cueh
```

Enter the password when prompted

Connecting in the rest of the Uni

The address of the server is: **zaphod.coventry.ac.uk**

Therefore the command to use SSH on the command line will be.

```
ssh level0@zaphod.coventry.ac.uk
```

Enter the password when prompted

Chapter 2: Hints and Tips

This next section gives some hints and tips for the task, (and for Uni in general). As well as taking my advice, talk to the second and third years. There are many "I wish I knew that then" story's that they can share.

Keep a Notebook

You will get bored of me/us saying this, but I will repeat it again **KEEP BLOODY NOTES**.

OK, you may break into the system, but your client is not going to be happy with a report saying "Yeah, I hit it with MSFVenom, and it fell over". Keeping track of how you achieved the exploit is vital for writing the final report so you get paid.

More importantly, it helps you remember *how* you exploited something, and *what* you were thinking when you did it. I keep notes on all of the CTF's I have done, and its incredibly useful when you recognise an exploit you have seen before, but cant remember the details.

> I don't really care what format, chose one that works for you. Some like Pen and Paper, some whatever travesty Microsoft wants you to use. Personally, I use text files, formatted with Markdown, and stashed on GitHub.

While we are at it, Cheat Sheets are also an *Excellent Idea*, stick all the commands you use regularly for tasks in a separate file, and it will help in the future. If you feel the need to be inspired, take a look at the Red Team Field Manual (its only a couple of quid, and is worth every penny).

Practice, Practice, Practice

With most things in life the more you do it, the better you become, (there are always exceptions). Keep using new skills, play with the command line, take part in CTF competitions. While the technical skills are important, most of the time I find its the way people think about the problem that is Vital. The more you do it, the easier it becomes to make the link between problem and solution.

If you play things like Hack the Box, go and read writups of earlier machines, sometimes seeing how others approach the problem is useful.

Get Permission before Penetration

One of the most important points here, and something that could keep you out of a world of hurt.

While as *Ethical* hackers, its our job to break into computer systems, we need to consider the Legal, Moral and Ethical factors around our work. In a purely Legal sense the Computer Misuse Act has some pretty severe penalties for gaining (or attempting to gain) unauthorised access to a computer system.

The rule you should follow is **Get permission from all involved parties before attempting to compromise a system** (a good rule for life as well as the course). At the end of the day, while getting permission may be a pain, it will keep you out of trouble.

So what does this mean for the course:

- If you are asked to break something as part of the course, then you have implicit permission
- You DO NOT have permission to attack any university Infrastructure
- If (and only if) I have released the bug bounty for the lab, then things like the Windows Image are fair game. **However**, any lab infrastructure is not (so the Imaging server, FTP, DNS etc) We rely on this to deliver the materials, so don't interfere with it.
- If you want to try something, **ASK**, most of the team are geeks and are super interested in security (i mean its our job), we can support you and make sure you will not accidentally occur the wrath of IT Services.
- The Flip side of this is that many people outside of the course are not security specialists. We have had instances of people wanting to kick students out for using NMAP outside of the labs (as its an "Hacking tool"). If you ask, we can make sure the correct people are informed, and avoid these sorts of problems.

256 Shades of Grey

The morals and ethics of compromising something without the permission of any involved parties are a hot topic. Often we hear terms such as "Grey Hat" and "Hacks for Good, but without permission". I would show some caution before taking this approach, as there have been cases of people prosecuted, and convicted for this.

There are some excellent bug bounty programs that may let you play with live systems, without the legal risk (although pay close attention to the Terms and Conditions, and make sure you stay within the boundaries that have been set by the owners).

Don't Be a Cheat!

Cheating sucks. While in this instance it might not seem to matter if you grab someone elses solution so you get "the win", (unlike formal assessment where if you are caught you may be kicked out of the Uni) you are missing the point. Part of the learning process is the journey, yes whacking your friends flag in the submissions box may make you feel great for having "completed" the challenge, but a lot of the course builds on actually being able to do things. When you can't SSH in the second year Being told to RTFM, really sucks. Work through the challenges, and if you get stuck ask for help, its much better for you.

> On a related note, part of the Fun for CTF challenges is solving the problem. I know you can find Flags for things like HTB online, but again it kind of defeats the point. Part of the fun of these things is the hard work, dedication, frustration, then the elation when everything works and you crack it, (or even the discussion down the pub about the (either) "Bloody simple thing I missed", "Absolute madness of the challenge" (such as hiding a damn bootloader inside an ASCII art Christmas's tree), when the competition is over and the writups come out.

Background Reading, and looking for other resources.

Read around the subject, go learn about the a set of commands for something, or learn what the Operating system is *actually doing* when you run a command. It will help you conceptualise the problems you face, which in turn makes it much easier to think about them. There are a load of great resources and blogs available for free online, so go make use of them.

RTFM

You may notice that some levels do not provide *all* the information you need to complete the task. Welcome to the world of "Background Reading", you are strongly encouraged to do your own research on the topics we introduce, not only is it part of University life but most hacking problems wont have an answer on stack overflow.

Ask for Help, and Talk to your peers.

Don't be afraid of asking for help, Everyone has different skill levels, and interests.

Hacking is such a wide ranging topic that once you get past the intermediate stage, you need to specialise (as spending all your time hacking is not good for you). As you go through Uni you will identify people with specialities like the "Windows Admin", "Linux Fan", "Web Hacker" or the "REENIGNE". If you are polite, (and have done at least some of the background reading first) and they can help, that's an excellent resource.

A Note on Background Reading

NOTE: The doing the background reading part is important. The difference in the response to questions like:

- "I don't understand Linux, Help Me!!!"
- "I get permissions but I don't understand SUID, could you explain how it can let me run stuff as a different user?"

Can be blunt. Don't expect to get the answer without effort. Go ask the first one if you want to see a "Live" version of a stack overflow flame war

ALSO: If you are the one of the above, reward those questions, It may take some of your time, but in all honesty, teaching someone this stuff is really rewarding, and you are likely to strengthen your understanding, just by explaining it to someone else.

Further Reading

- **Computer Misuse Act**
- `https://www.cps.gov.uk/legal-guidance/computer-misuse`
- **Linux Survival Guide**
- `http://matt.might.net/articles/basic-unix/`
- **Over the Wire** Excellent resource that "inspired" this work. Give Bandit a go,
- `http://overthewire.org/wargames/`
- **Mobaterm**
- `https://mobaxterm.mobatek.net/`
- **Putty**
- `https://www.putty.org/`