# ITRI625 - Computer Security II

# Metasploit Project Documentation

Affaan Muhammad - 33016763

Joshua Esterhuizen - 30285976

Due: October, 19th 2021

# Contents

# Section 1

# Metasploit

## 1.1 Introduction

The growth and widespread adoption of technology and the internet can be compared to a double-edged sword. While the globe is more interconnected than ever, this almost unanimous adoption of technology has brought with it some issues and problems of its' own.

One such major issue is the increased security risk people undertake by participating in the use of technology and the internet. Before the most risk you would be in was if you had dropped important documents while nowadays an attacker could access this information through some nefarious mean without even alerting you or the targeted institution. The simplest was to guard against these on a large scale without sacrificing the comfort of using technology or the internet is to develop and put in place security measures to block these kinds of attacks.

The Metasploit framework was developed to aid in this process by providing a structured and contained way to attempt penetration on a system to test the security measures and their effectiveness. This is helpful for both countering security in the wrong hands as well as aiding in improving the security of organisations.

## 1.2 What is Metasploit?

Before we can describe the ways that Metasploit can help organisations and how it counters security in the wrong hands, we need to understand what Metasploit is and how it functions. Metasploit and its' framework were originally designed and developed as a tool for security experts in various fields such as network security, security administrators, product vendors and any other security experts to use within their own field according to the specific needs of each.[5]

Other than that, Metasploit is describes as a tool that collectively combines exploits into one central hub for security experts and researchers or alternatively as a project that contains information pertaining to security vulnerabilities and aids in the penetration testing of a system as well as the development of an intrusion detection system.[9, 5]

Penetration testing is a method of identifying certain vulnerabilities within a system be it a computer, network or website. As a process, it includes gathering data on the system to determine where vulnerabilities may lie and then attempt to exploit them to test the measures put in place.[9]

Metasploit was originally equipped with many modules that can be used. The four main modules include:[9]

- Exploit module which allows for a means to attack a compromised system,
- Payload which allows for a means for attackers to gather data by interacting with the target,
- Auxiliary module, which is equipped to test, scan and recon the target and
- Encoders module which is used for evading firewall and anti-virus tools.

It should also be noted that since Metasploit is an open source, meaning the code to how it was built is publicly available for download, this allows users and security experts to develop their own modules if the need for then is required. [9]

There are also two other notable items that are coupled with Metasploit – those being Meterpreter and Msf-Venom. Meterpreter is a powerful payload that came to Metasploit some time after it was developed. It is an advanced dynamically integrated payload that is stored entirely within the memory of a system. [5]When a vulnerable system is found and consequentially infected with this payload via any exploit, it establishes a connection between the infected and attacker systems through a client side, the infected system, command which bypasses most firewall restrictions.[5]

Meterpreter is also a powerful payload for another reason - it is very elusive and often goes by undetected by most security experts.[5]

Meterpreter also comes with a command line style interface that has various commands such as getting system information, interacting with peripherals on the infected system and making use of the infected systems command line or power shell.

Msf-Venom on the other hand is not a payload but a Metasploit standalone payload generator. It is the updated replacement of msfpayload and msfencode.[9]

It allows a user to generate any type of shellcode or payload that will affect a platform of their choice - such as on Linux, Windows or Android to name a few.[9]

## 1.3 How it counters security in the wrong hands

"Security in the wrong hands" is likely referring to people or organisations that are making use of security measures to halt, attack or generally inconvenience others. As such, Metasploit and its' accompanying modules are a tool that will counter these types of attacks.

The clearest way that Metasploit can be used in this capacity is that it directly provides the tools to infiltrate and attack the system to remove the security measures in place and as such counters the given attack. Additionally, it can also be used to study the system itself to find any vulnerabilities and then generate payloads and exploits that match the systems security measures with Msf-Venom.

Metasploit allows for this with several keywords within the framework - namely those being:

- Search
- Use
- Exploit and Run

The search keyword allows for a Metasploit user to comb through the massive library of modules Metasploit provides and quickly and easily find the appropriate payload they require.[9]

The use keyword allows for a Metasploit user to implement the particular module they want to and begin the exploitation process.[9]

The exploit and run keywords allow a Metasploit user to activate their choice of payload and module and begin the actual infiltration of a vulnerably system. There are two keywords for this particular action.[9]

However, Metasploit can also be used by security experts within an organisation to test their internal security systems and measures that they have in place.

## 1.4 How it improves the cyber security of an organisation

Security experts and system administrators within organisations have begun taking the approach of viewing their control measures as an ever-evolving system that needs constant attention as it is a matter of when the measures will fail and not if they will fail.[2]

These experts and administrators can make use of Metasploit to test their systems and

find any vulnerabilities in a controlled environment before they are exploited by any malicious attacker. This is a needed aspect of the security department of any organisation that wished to keep their data and information safe as one of the biggest issues to solve with exploits is how they evade detection. This is typically done by avoiding detection of an anti-virus or anti-malware program through the use of [2, 5]:

- Manual Binary Editing
- Polymorphic code

Most anti-virus programs have been ineffective at detecting the payloads distributed by Metasploit and as such making use of these exact payloads to further secure a system is ideal.[5]

However, before being able to update a security system to detect these payloads, the means of evasion must be understood. Manual binary editing is the process by which the malware in question changes a particular signature to avoid detection by signature-based security programs.[2]

Due to this, the security experts could modify the means by which these applications check for signatures – as one suggestion is to sub-divide the files being scanned and checking if there is a partial match to a known malware or virus signature.[2]

The above method, however, may not work on the second means of evasion -polymorphic code. A virus or piece of malware with this specific trait changes each time it is run and therefore has a different signature each time - avoiding signature-based detection entirely. [2]

Therefore using payloads from Metasploit with this characteristic allows for system administrators and security experts to efficiently test the intrusion detection systems and any heuristic-based anti-malware or anti-virus programs.[2]

Another use is that Metasploit can also be used to test the security measures of websites and web applications.[8]

From this section, it is clear that Metasploit can be used to strengthen the security measures put in place by an organisation by allowing the security experts and system administrators to firstly identify what vulnerabilities the system has, then exploit it to understand how the exploit uses the vulnerability and then find a means to patch this vulnerability.

## 1.5 Conclusion

Metasploit is shown to be a powerful tool for security experts as it allows for a controlled means to study system vulnerabilities and exploits while aiding in the process of strengthening the systems themselves as well as the programs used to safeguard the systems.

# Section 2

# Installation and Setup

For further details have a look at our "DIY" blog page for the setup that was done in this project. It is located at the following link:
https://ITRI625.github.io/post3.html

The following tools were utilised:

- Oracle VirtualBox
- Vulnerable Windows 7 .iso image
- Vulnerable Android x86_64 .iso image
- Kali Linux .iso image

## 2.1 Blog

The blog we created was hosted on GitHub Pages. The link to the blog is:
https://ITRI625.github.io

More information on GitHub Pages can be found at: https://pages.github.com/

The template for the blog was acquired from:
https://startbootstrap.com/theme/clean-blog

GIFs on the Blog were sourced from https://tenor.com/ and https://giphy.com/

Additionally, the screenshots taken were from VMs we implemented in VirtualBox and other images for the headers were sourced from Google Images.

# Section 3

# Scenario 1: Android exploit

## 3.1 Overview & Carrying out the exploit

The scenario is described as follows:

> *A person (that will be known as the victim in further discussions), opens a SMS received on their mobile device. This SMS has a malicious link embedded in it, or has a link that redirects to a malicious site. The link automatically downloads and runs an Android application package also known in other terms as an APK file. This APK file is what the Android Operating System uses to install applications. This malicious Application, once installed gives the Hacker (known as a perpetrator in an attack) full control over the device to outside parties, including the perpetrator. From here on out, the perpetrator has the victim in the palm of their hands, and can do what every they like to the device as well as carry out further attacks against the victim.*

The following resources were cited during this process.[3, 7, 10, 1, 11, 4, 6]

For further details have a look at our "Exploit 1" blog page for this exploit. It is located at the following link:
https://ITRI625.github.io/post1.html The following basic procedures/steps were followed:

1. Install Oracle VirtualBox
2. Load Kali Linux image into VirtualBox
3. Load vulnerable Android x86_64 image into VirtualBox
4. Carry out the exploit
5. Carry out post exploitation

## 3.2 Countermeasures

The following advice can be given to users to prevent this type of exploit on their Android devices.

- Do not use open Wi-Fi networks.
- Disable the `"Install Apps from Unknown sources"` option in your device settings.
- Only install verified applications from the Google Play store or other trusted stores such as F-Droid.
- Do not open files that are sent to you from unknown parties.
- Install and keep updated an Anti-Malware tool on your device.
- Install and keep updated an Anti-Virus software on your device.
- If possible, pay for a VPN service for your device.
- Do not carry out sensitive processes on your device if it is not secured or you do not have a VPN.
- Do not open any correspondence you receive on your device that is from an unknown party.
- Keep you device updated at all times, especially security updates/patches.
- Keep your applications on your device updated at all times and if possible disable auto-updates. This will allow you to keep track of what is updated and when.
- Delete any application on your device if you have not used it in the past month (This excludes your built-in applications).
- Disable Background processes (for applications) on your device.

# Section 4

# Scenario 2: Windows exploit

## 4.1 Overview & Carrying out the exploit

The scenario is described as follows:

> *Windows 7 had all support for the operating system halted in January of 2020. Windows 7 was certainly a user favourite and the end of support saw quite a few users disgruntledly switch to either Windows 8 or 10 to continue to get security focused support and patches. At this point, the support for windows 7 has been defunct for almost 2 years and as such many new exploits have likely surfaced. Nonetheless, even before Windows 7 met its end-of-life there were still many exploits that could be performed on the system that were sequentially fixed in the new versions of Windows. One such vulnerability present in Windows 7 is that the default version of Windows Media Centre will execute any code saved as a ".mcl" file.*

For further details have a look at our "Exploit 2" blog page for this exploit. It is located at the following link:
https://ITRI625.github.io/post2.html

The following basic procedures/steps were followed:

1. Install Oracle VirtualBox
2. Load Kali Linux image into VirtualBox
3. Load vulnerable Windows 7 image into VirtualBox
4. Carry out the exploit
5. Carry out post exploitation

## 4.2 Countermeasures

The most useful countermeasure to this, and typically, any security related vulnerability is user vigilance. A user should be aware of what files, services, websites, etc. should typically look like in their daily usage a computer system and, as such, should be able to identify when one of these, and in this case a file, looks suspicious and should be scanned via some program before using or opening it.

Furthermore, it is also important since this exploit made use of a .mcl and .exe file that a user makes use of some form of antivirus or antimalware software and in this particular case, ones that are signature based. The reasoning for this is because the executable in this scenario is one that has been created and is typically used in a large scale.

It is also vital to understand the exploits you are vulnerable to. In the tutorial we demonstrated that the file could be run and Windows only alerts that it comes from an untrusted, meaning un-certified, source which nowadays is something that happens with most applications as the typically programmer does not have access to particular licencing on their applications. In this case, having a more rigorous firewall or settings for it may have restricted the reverse_tcp from functioning.

Apart from this, the main fix to this particular issue was to install a patch provided by Microsoft on their site, and as such having some sort of patch management system is also vital to users. This could be simply regularly checking each application download page for the newest version or patch. However, there are applications and programs that automate this process and keep all your programs up to date. It should also be noted that most programs tend to offer a "Check for updates" feature when installing the program and likely also in the settings after the fact.

# Section 5

# Closing remarks

## 5.1 Reflection & Work Consensus

The group for this project consisted of 2 members namely:

- Affaan Muhammad - 33016763
- Joshua Esterhuizen - 30285976

Below is our reflection that was carried out after the completion of the project.

**Reflection**

Joshua Esterhuizen had the following to say:

> *Working on this project has taught me several different things both within and outside of computer security. The most notable to me personally would be the fact that GitHub allows each user to create and then host a single static website. This is what was used to host the blog section of this project but will also be used by me in a personal capacity for a personal website in future.*
>
> *Apart from this, seeing the sheer amount of exploits that Metasploit has available to use is a bit disconcerting as it covers a wide array of different machines, programs and means of execution and has definitely made me that much more critical of everything I do online.*
>
> *This project also allowed me the chance to learn how to use HTML and CSS when writing the blog, which also forced me to switch up my usual writing style for one that is more colloquial for the general public instead of assuming the people reading the writing are already knowledgeable on the topics.*

Affaan Muhammad had the following to say:

> *The Computer Security II module (ITRI625) was an enjoyable experience overall. The practical aspects were an added bonus which helped strengthen the theoretical knowledge that was learned during the year.*
>
> *Additionally it was also enjoyable to once again delve into details of Operating Systems, as well as Computer Networking which was done in the final year of our undergraduate degree. So in essence during the Honours year we built upon the foundations laid in the undergrad years.*
>
> *It was tremendous fun to once again do Web Development and implementing static pages using GitHub Pages.*
>
> *A few mishaps that have to be mentioned is the tricky work-around that was done to implement NAT Networks to allow communication between virtual environments. Any other issues that came up were solved as soon as possible and being able to navigate and sift through error was key to the success for this project.*
>
> *Using the Kali Linux operating system, the command line, debugging, and scripting was enjoyable as always and helped in adding to our knowledge base.*
>
> *Additionally, good time management and excellent communication was vital to the added success of this project.*

**Work Consensus**

All members in the group contributed to this project and a 50/50 balance between work allocation was kept. Below is a table showing how the work was divided in this project amongst the 2 members i.e. Affaan & Joshua

| Affaan | Joshua |
|---|---|
| Scenario 1 | Scenario 2 |
| Blog | Blog |
| Testing | Testing |
| Bug fixes | Bug fixes |
| Proofreading | Proofreading |
| Documentation | Metasploit Literature Review |

# Section 6

# Additional readings and miscellaneous information

CitizenLab located in Toronto, Canada is responsible for actively testing new threats, and exploits. Their site is located at the following url:
https://citizenlab.ca/
You can find additional information and news of current events in the cyber security space.

Heimdal Security headquartered in Denmark are a company which creates different suites for your cyber security needs. They also offer interesting resources such as whitepapers, articles, blogs, data sheets, case studies etc. Their site is located at the following url:
https://heimdalsecurity.com/

Additionally, an extensive list of cyber security resources can be found at:
https://www.cyberdegrees.org/resources/the-big-list/
They include the following:

## 6.1 News and Opinions

- Ars Technica – Risk Assessment
- CIO Security
- CSO Online
- Dark Reading
- Guardian Information Security Hub
- Homeland Security News Wire – Cybersecurity
- Infosecurity Magazine
- Naked Security
- SC Magazine
- SecureList
- SecurityWatch
- Threat Level
- ThreatPost

## 6.2 Blogs

- Google Online Security Blog
- InfoSec Resources
- Krebs on Security
- Microsoft Malware Protection Center Blog
- Schneier on Security
- Security Bloggers Network
- Terebrate
- Threat Track Security Labs Blog
- Veracode Blog
- Zero Day Blog

## 6.3 Other useful websites

- UTPA Center of Excellence in STEM Education
- CERIAS: Tools and Resources
- CVE: Common Vulnerabilities and Exposures
- Information Security Stack Exchange
- Infotec Pro
- ISC: Internet Storm Center
- National Centers of Academic Excellence (CAE) in Information Assurance (IA)/Cyber Defense (CD)
- OVAL: Open Vulnerability and Assessment Language
- Scholarship Opportunities
- US-CERT
- U.S. Department of Homeland Security – Cybersecurity

## 6.4 Organisations

- ACM SIGSAC: Special Interest Group on Security, Audit and Control
- ASIS International
- CSA: Cloud Security Alliance
- DC3: Defense Cyber Crime Center
- HTCIA: High Technology Crime Investigation Association
- ISF: Information Security Forum
- ISSA: Information Systems Security Association
- NICCS: National Initiative for Cybersecurity Careers and Studies
- NSI: National Security Institute
- NW3C: National White Collar Crime Center
- OWASP: Open Web Application Security Project
- SANS
- Science of Security Virtual Organization

## 6.5 Training

- Damn Vulnerable Web Application (DVWA)
- Evolve Security Academy
- HackThisSite (HTS)
- Metasploitable
- Mutillidae
- NATAS
- National Institute of Building Sciences
- SecureSet
- SlaveHack

## 6.6 Security Groups

- AFCEA Chapters
- CSA Chapters
- IEEE Technical Chapters
- InfraGard Local Chapters
- ISACA Local Chapters
- $(ISC)^2$ Chapter Program
- ISSA Chapter Directory
- OWASP Chapters Program

## 6.7 Contests and Competitions

- CSAW Capture the Flag (CTF)
- DEF CON Contests
- ESC: Embedded Security Challenge
- NCCDC: National Collegiate Cyber Defense Competition
- NCL: National Cyber League
- Panoply
- Pitcoctf
- Pwn2Own
- Pwnium
- SANS NetWars
- U.S. Cyber Challenge

## 6.8 Conferences

- ACM CCS: ACM Conference on Computer and Communications Security
- ACSAC: Annual Computer Security Applications Conference
- Asiacrypt/Crypto/Eurocrypt
- Black Hat
- BSides
- CanSecWest
- CSAW: Cyber Security Awareness Week Conference

- DeepSec
- DEF CON
- DerbyCon
- Hack.lu
- Hacker Halted
- The Hackers Conference
- Hackito Ergo Sum
- HITBSecConf: Hack In The Box Security Conference
- ICMC: International Cryptographic Module Conference
- IEEE Symposium on Security and Privacy
- NDSS (Network and Distributed System Security) Symposium
- NSPW: New Security Paradigms Workshop
- Nullcon
- RSA Security Conference
- SANS CDI: Cyber Defense Initiative
- S4: SCADA Security Scientific Symposium
- Secure 360
- SecureWorld Expo
- ShmooCon
- SIN: International Conference on Security of Information and Networks
- SOURCE Conference
- Swiss Cyber Storm
- Thotcon
- TROOPERS IT Security Conference
- USENIX Security Symposium
- VB: Virus Bulletin Conference

# Bibliography


[1] *Binary Payloads*. URL: https://www.offensive-security.com/metasploit-unleashed/binary-payloads/.

[2] Peter Casey et al. "Applied comparative evaluation of the metasploit evasion module". In: *2019 IEEE symposium on computers and communications (ISCC)*. IEEE. 2019, pp. 1–6.

[3] *Command-line Flags Chapter 4. Port Scanning Overview*. URL: https://nmap.org/book/port-scanning-options.html.

[4] *Manage Meterpreter and Shell sessions*. URL: https://docs.rapid7.com/metasploit/manage-meterpreter-and-shell-sessions/#view-available-meterpreter-shell-commands.

[5] Carlos Joshua Marquez. "An analysis of the ids penetration tool: Metasploit". In: *The InfoSec Writers Text Library, Dec* 9 (2010).

[6] *Meterpreter Basic Commands*. URL: https://www.offensive-security.com/metasploit-unleashed/meterpreter-basics/.

[7] *Meterpreter: Security encyclopedia*. URL: https://www.hypr.com/meterpreter/.

[8] Indraneel Mukhopadhyay, S Goswami, and E Mandal. "Web penetration testing using nessus and metasploit tool". In: *IOSR Journal of Computer Engineering* 16.3 (2014), pp. 126–129.

[9] Sudhanshu Raj and Navpreet Kaur Walia. "A Study on Metasploit Framework: A Pen-Testing Tool". In: *2020 International Conference on Computational Performance Evaluation (ComPE)*. IEEE. 2020, pp. 296–302.

[10] *What is Meterpreter ? - Security Wiki*. Aug. 2021. URL: https://doubleoctopus.com/security-wiki/threats-and-tools/meterpreter/.

[11] Mohammed Zain. *How It Works: Reverse_tcp Attack*. Apr. 2020. URL: https://medium.com/@mzainkh/how-it-works-reverse-tcp-attack-d7610dd8e55.