Cybersecurity

Linde Nouwen

Karel de Grote Hogeschool

CyberSecurity Exploitation

Karel de Grote Hogeschool

Exploitation

- General hacking mechanisms
- Methodical Approach
- Frameworks
- Defence

Exploitation

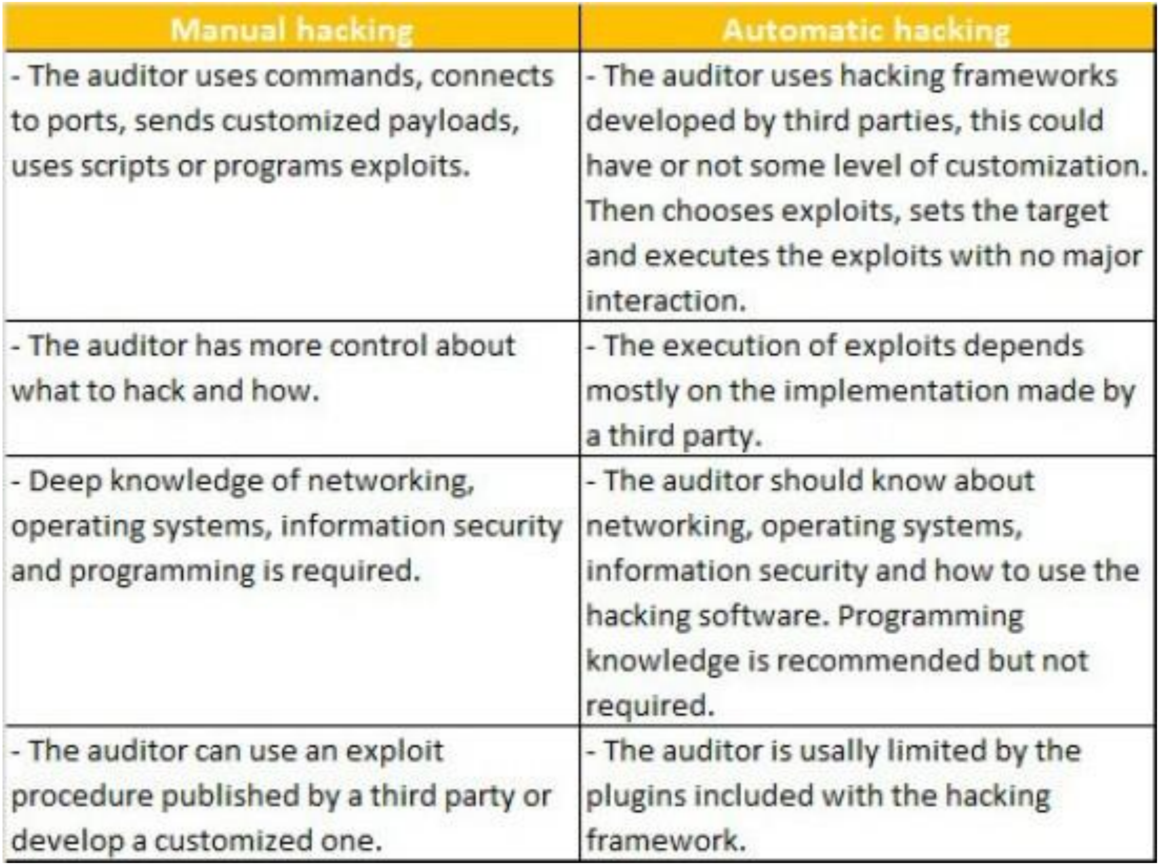
General Hacking Mechanisms

Exploitation

- General Hacking Mechanisms:
  - Manual Hacking
  - Automatic Hacking

Exploitation

- Hacking Mechanisms:
  - Manual Hacking
  - Automatic Hacking



Manual hacking	Automatic hacking
- The auditor uses commands, connects to ports, sends customized payloads, uses scripts or programs exploits.	- The auditor uses hacking frameworks developed by third parties, this could have or not some level of customization. Then chooses exploits, sets the target and executes the exploits with no major interaction.
- The auditor has more control about what to hack and how.	- The execution of exploits depends mostly on the implementation made by a third party.
- Deep knowledge of networking, operating systems, information security and programming is required.	- The auditor should know about networking, operating systems, information security and how to use the hacking software. Programming knowledge is recommended but not required.
- The auditor can use an exploit procedure published by a third party or develop a customized one.	- The auditor is usally limited by the plugins included with the hacking framework.

Exploitation

- General hacking mechanisms
- Methodical Approach
  - Password Cracking
  - Privilege Escalation
  - Executing Applications
  - Hiding files, covering tracks and concealing evidence
  - Expand the attack…
- Frameworks
- Defence

Exploitation

Password cracking

Exploitation

- Password Cracking
  - Enumeration = you got the username
  - Now you try to get the password
  - Dualism in passwords:
    - Easy to remember
    - Not easy to guess or break
  - “Don’ts”:
    - All numbers
    - All letters
    - All same case
    - Proper names
    - Dictionary words
    - Short

Security/Convenience Dilemma

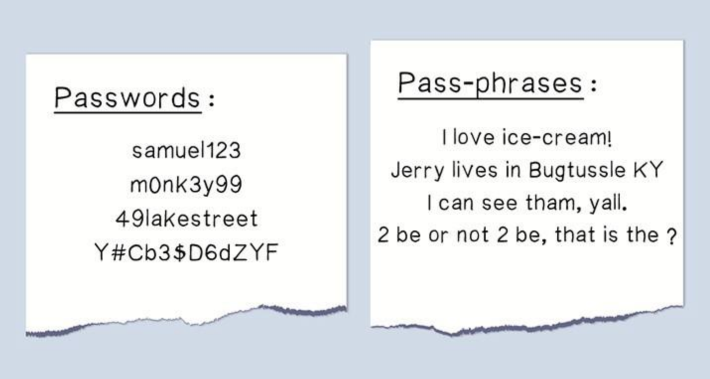
Exploitation

- Password Cracking/guessing
  - Dictionary Attacks.
  - Brute-Force Attacks
  - Hybrid Attack
  - Syllable Attack
  - Rule-Based Attack
  - Passive Online Attacks
  - Active Online Attacks
  - Offline Attacks
  - Nontechnical Attacks

Exploitation

- Password Cracking

Pasword-construction:



Passwords :

samuel123
m0nk3y99
49lakestreet
Y#Cb3$D6dZYF

Pass-phrases :

I love ice-cream!
Jerry lives in Bugtussle KY
I can see tham, yall.
2 be or not 2 be, that is the ?

Dictionary Attacks

An attack of this type takes the form of a password-cracking application that has a dictionary file loaded into it. The dictionary file is a text file that contains a list of known words up to and including the entire dictionary. The application uses this list to test different words in an attempt to recover the password. Systems that use passphrases typically are not vulnerable to this type of attack.

Brute-Force Attacks

In this type of attack, every possible combination of characters is attempted until the correct one is uncovered. According to RSA Labs, “Exhaustive keysearch, or brute-force search, is the basic technique for trying every possible key in turn until the correct key is identified.”

Exploitation

- Password Cracking

Pasword-construction:

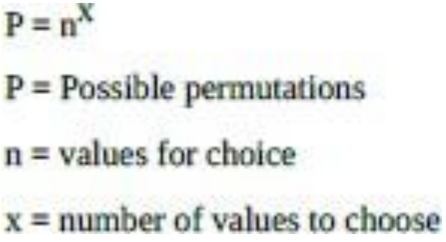
Hybrid Attack This form of password attack builds on the dictionary attack but with additional steps as part of the process. In most cases, this means passwords that are tried during a dictionary attack are modified with the addition and substitution of special characters and numbers, such as *P@ssw0rd* instead of *Password.*

Syllable Attack This type of attack is a combination of a brute-force attack and a dictionary attack. It is useful when the password a user has chosen is not a standard word or phrase.

Rule-Based Attack This could be considered an advanced attack. It assumes that the user has created a password using information the attacker has some knowledge of ahead of time, such as phrases and digits the user may have a tendency to use.

Exploitation

- Something on “Brute Force”
  - Formula:



$P = n^X$

P = Possible permutations

n = values for choice

x = number of values to choose

  - Normally used:
- Uppercase letters: A-Z
- Lowercase letters: a-z
- Numbers: 0-9
- Symbols: ~`!@#$%^&*()_-+={[}]|\:;"'<,>.?/
  - count =32

Exploitation

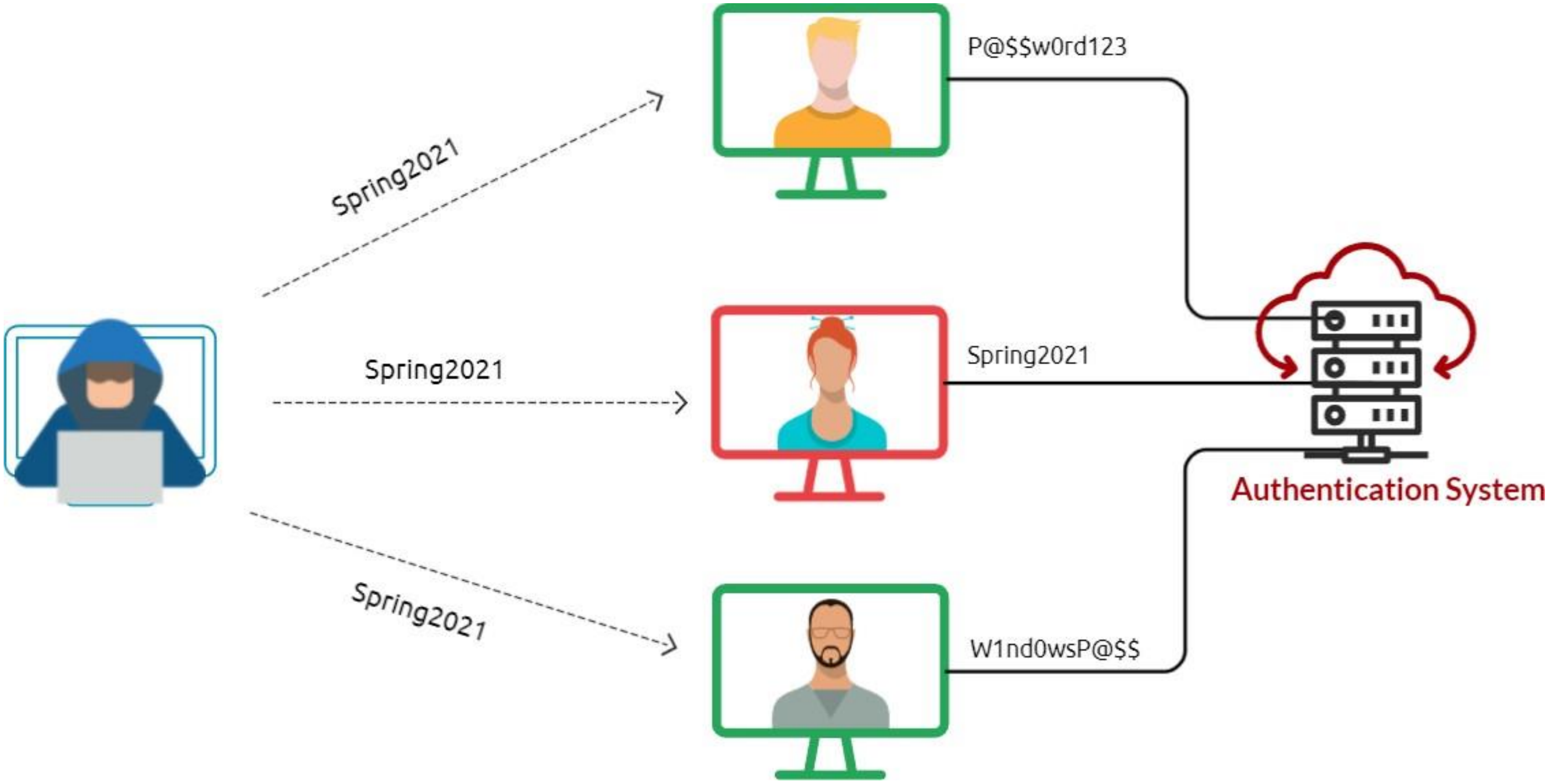
- Something on “Brute Force”
  - Example1:
    - Password = 2 numeric characters
    - Formula: P=10^2=100 combinations
      - Evaluation = can be done
  - Example2:
    - Password = 10 characters (letters and numbers)
    - Formula: P=(26*2+10)^10 = 8,39299E+17 combinations
      - Evaluation: impossible?
  - Example3:
    - Password = 20 characters (letters + numbers+ symbols)
    - Formula: P=(26*2+10+32)^20=2,90106E+39
      - Evaluation: impossible?

Exploitation

- Something on “Brute Force”
  - Time is limiting factor in cracking
  - Unless you “know something” (= shrink possible combinations)
  - Use wisely…

"Reverse" Exploitation

- Password Spraying



Exploitation

Password cracking
Passive & Active online cracking…

Exploitation

- Password Cracking
  - Passive Online Attacks
    - Packet Sniffing
    - Man-in-the-Middle
    - Replay
  - Active Online Attacks
    - Guessing
    - Trojan, Spyware or keylogger
    - Hash injection

Not considered real cracking, cracking is usually done offline using a compromised hash

Exploitation

- Password Cracking
  - Passive Online Attacks
  - Packet Sniffing
    - Network card in “promiscuous” mode = get all packets (also for other MAC addresses)
    - Packet sniffer captures packets
    - Contents are examined for readable or poorly protected passwords (FTP, Telnet, SMTP, …)

Exploitation

- Password Cracking
  - Passive Online Attacks
  - Packet Sniffing
    - Difficult with modern switches (Hubs are OK)
      - » Options:
        - Retrieve from switch/router level…
        - Attack switch
        - Retrieve from end-device
        - Deceive end-device (MITM attack)
    - Wireless still OK

Exploitation

- Password Cracking
  - Passive Online Attacks
  - Packet Sniffing
    - Switch-attacking:
      - Objective: make it behave like a hub (= replicate packets to all ports)
      - Technique: mac flooding.
      - Software that generates one frame after another, with fake source MAC addresses generated randomly
      - Switch’s MAC table begins to grow in size
      - Device memory gets full.

Exploitation

- Password Cracking
  - Passive Online Attacks
    - Packet Sniffing
  - Switch-attacking:
    - » Possibe Results:

1. switch responds deleting its MAC table, reversing its behavior to a hub → YES!
2. switch can´t support the load causing a momentary denial of service to the LAN
3. switch is secured against this attack and that as a result we get ourselves blocked from accessing the LAN and busted

Exploitation

- Password Cracking
  - Passive Online Attacks
  - Man-in-the-Middle
    - Third party puts itself in the middle of the communication of 2 other parties
    - Eavesdropping
  - Can be used for
    - Information gathering
    - Information altering
    - “Dual-side sniff” if needed = no detection
  - But …
    - Tricky to execute
      - » Utilities exist (SSL Strip, Burp Suite, Browser Exploitation Framework, …)

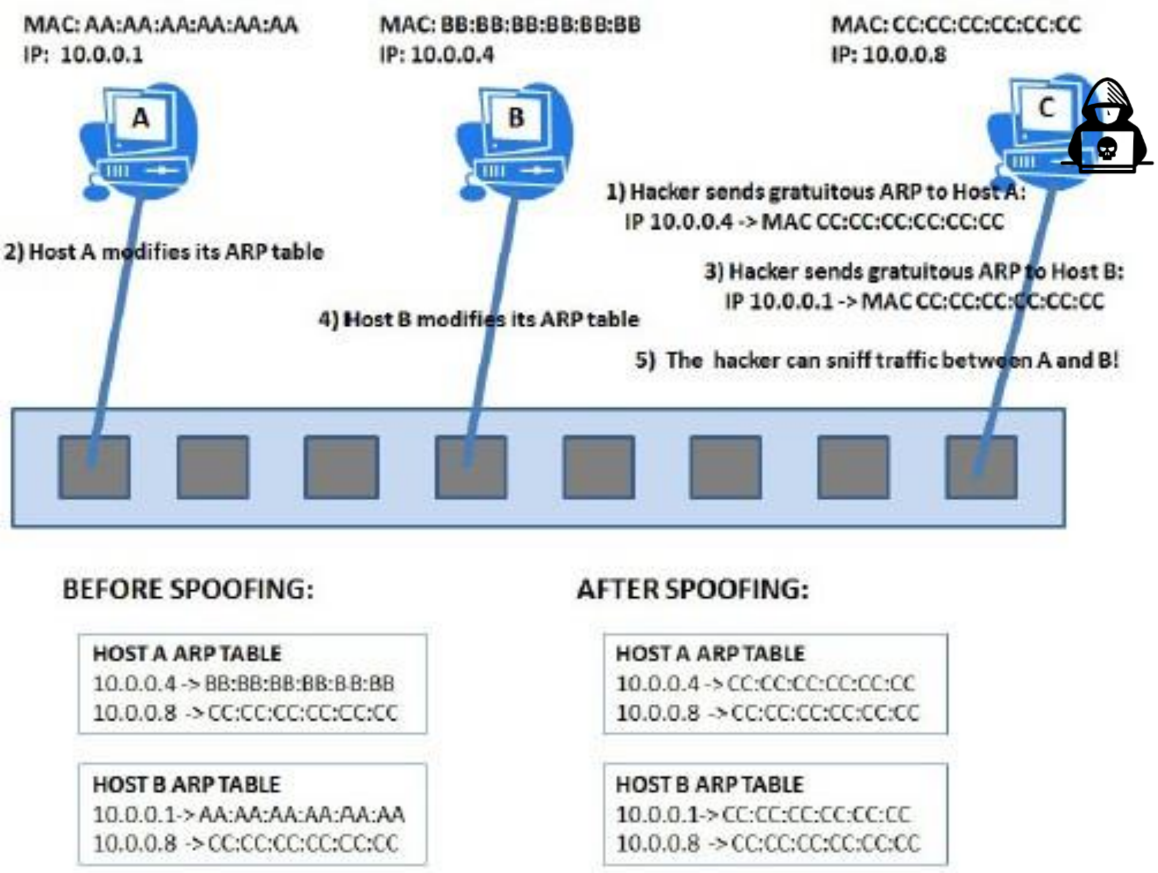
Exploitation

- From Spoofing to MITM attack
  - Simplest form = ARP spoofing
  - All systems have ARP tables (link IP to MAC)
  - Send (adapted) “gratuitous ARP” to target
    - saying that your MAC = the IP of the machine that the target should communicate with
    - Target will now send its packets to you
  - Receive the packets, record them and adapt them to follow normal route = only MAC (layer2) info gets changed
  - Put ARP of target back to original after attack = traffic gets back to normal. Avoid detection.

Note: 2-way if needed

Exploitation

- From Spoofing to MITM attack
  - Use ARP spoofing to setup MITM communication:
    - ARP



MAC: AA:AA:AA:AA:AA:AA
IP: 10.0.0.1

MAC: BB:BB:BB:BB:BB:BB
IP: 10.0.0.4

MAC: CC:CC:CC:CC:CC:CC
IP: 10.0.0.8

1) Hacker sends gratuitous ARP to Host A:
IP 10.0.0.4 -> MAC CC:CC:CC:CC:CC:CC

2) Host A modifies its ARP table

3) Hacker sends gratuitous ARP to Host B:
IP 10.0.0.1 -> MAC CC:CC:CC:CC:CC:CC

4) Host B modifies its ARP table

5) The hacker can sniff traffic between A and B!

BEFORE SPOOFING:

HOST A ARP TABLE
10.0.0.4 -> BB:BB:BB:BB:BB:BB
10.0.0.8 -> CC:CC:CC:CC:CC:CC

HOST B ARP TABLE
10.0.0.1-> AA:AA:AA:AA:AA:AA
10.0.0.8 -> CC:CC:CC:CC:CC:CC

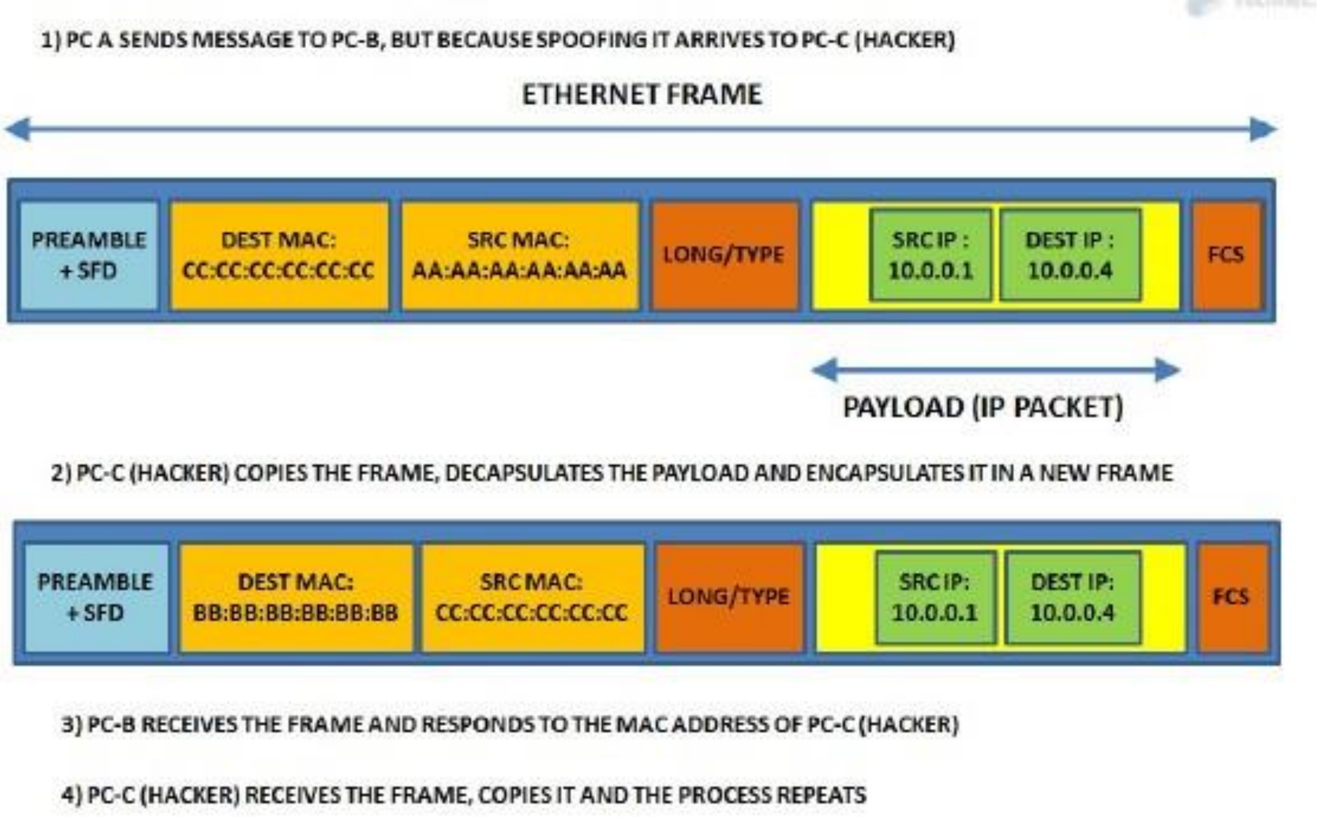
AFTER SPOOFING:

HOST A ARP TABLE
10.0.0.4 -> CC:CC:CC:CC:CC:CC
10.0.0.8 -> CC:CC:CC:CC:CC:CC

HOST B ARP TABLE
10.0.0.1-> CC:CC:CC:CC:CC:CC
10.0.0.8 -> CC:CC:CC:CC:CC:CC

Exploitation

- From Spoofing to MITM attack
  - Final piece of the puzzle for MITM attack.
    - IP-forwarding by hacker



1) PC A SENDS MESSAGE TO PC-B, BUT BECAUSE SPOOFING IT ARRIVES TO PC-C (HACKER)

PREAMBLE + SFD	DEST MAC: CC:CC:CC:CC:CC:CC	SRC MAC: AA:AA:AA:AA:AA:AA	LONG/TYPE	SRC IP : 10.0.0.1	DEST IP : 10.0.0.4	FCS

2) PC-C (HACKER) COPIES THE FRAME, DECAPSULATES THE PAYLOAD AND ENCAPSULATES IT IN A NEW FRAME

PREAMBLE + SFD	DEST MAC: BB:BB:BB:BB:BB:BB	SRC MAC: CC:CC:CC:CC:CC:CC	LONG/TYPE	SRC IP: 10.0.0.1	DEST IP: 10.0.0.4	FCS

3) PC-B RECEIVES THE FRAME AND RESPONDS TO THE MAC ADDRESS OF PC-C (HACKER)

4) PC-C (HACKER) RECEIVES THE FRAME, COPIES IT AND THE PROCESS REPEATS

Exploitation

- Password Cracking
  - Passive Online Attacks
  - Replay
    - Previously captured packets (with password which doesn’t need to be known) are resent
    - Authentication result is sent to hacker

Exploitation

- Password Cracking
  - Passive Online Attacks
    - Packet Sniffing
    - Man-in-the-Middle
    - Replay
  - Active Online Attacks
    - Guessing
    - Trojan, Spyware or keylogger
    - Hash injection

Exploitation

- Password Cracking
  - Active Online Attacks
  - Guessing
    - Manually
    - Software application (dictionary with variations)

Exploitation

- Password Cracking
  - Active Online Attacks
  - Trojan, Spyware or keylogger
    - Malware
    - Usually: keyboard sniffing or keylogging

Exploitation

- Password Cracking
  - Active Online Attacks
  - Hash injection
    - Extract the password hashes for users with high permission profiles
    - Use hash to log on to server/domain controller

So, …

- What is Hashing?
- What are Rainbow Tables

Exploitation

- Password Cracking
  - What is Hashing?
  - What are Rainbow Tables

Exploitation

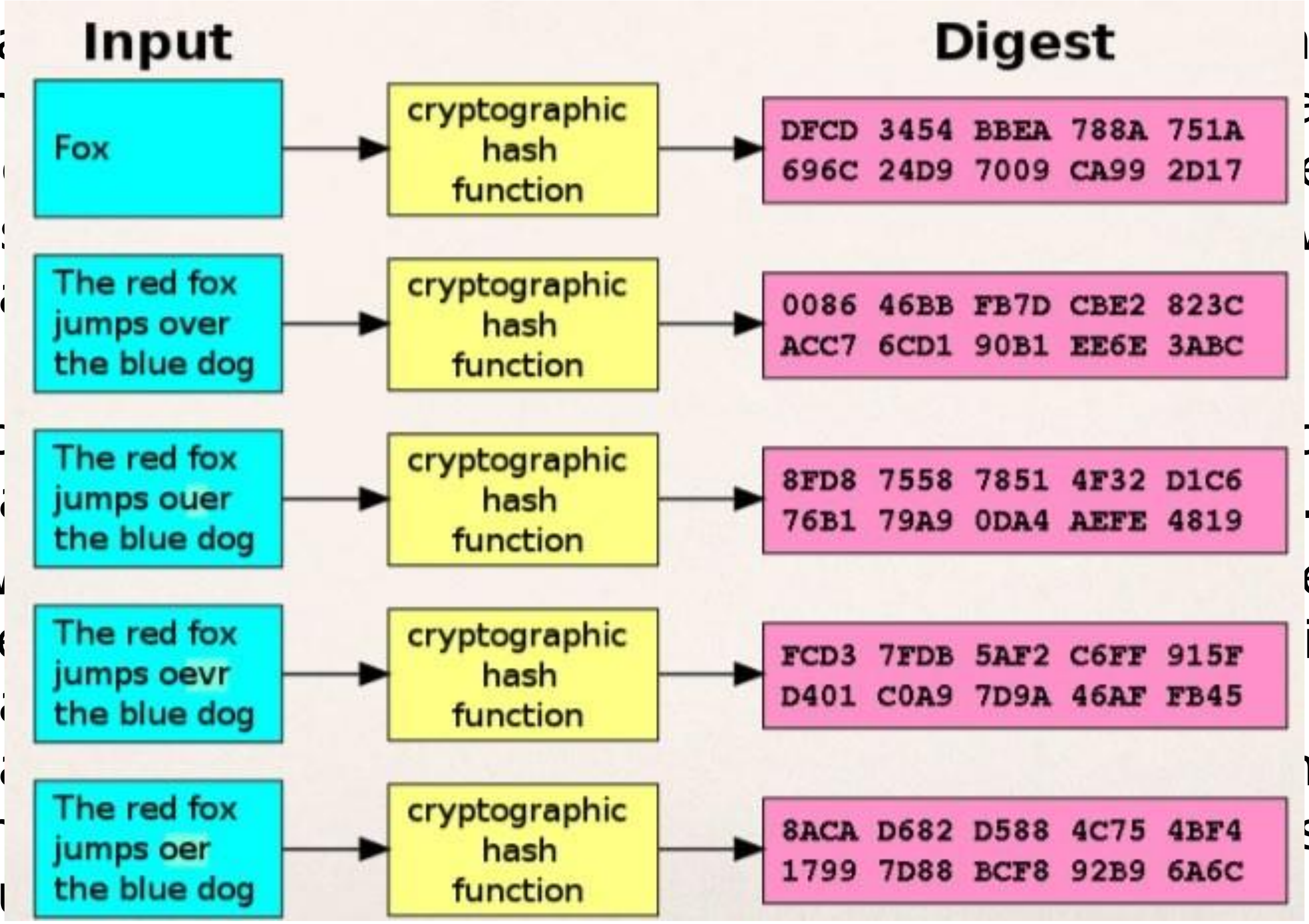
- Password Cracking
  - What is Hashing?
    - Passwords are not stored in clear text on a system in most cases because of their extremely sensitive nature. Because storing passwords in the clear is considered risky, you can use security measures such as password hashes
    - Hashing is a form of **one-way encryption** that is used to verify integrity. Passwords are commonly stored in a hashed format so the password is not in clear text.
    - When a password provided by the user needs to be verified, it is hashed on the client side and then this hash is transmitted to the server, where the stored hash and the transmitted hash are compared. If they match, the user is authenticated; if not, the user is not authenticated.

Exploitation

- Password Cracking
  - Hash:
    - Hash: one-way mathematical function to turn text of any size into result of fixed size
      - H(x)=y
      - H(z)=y
      - Must mean x=z
    - Size of text can’t be deduced from hash result
    - Principle:
      - Password is created.
      - System calculates hash and stores result(!) in db
      - Next time password is entered: hash algorithm runs and result is checked against db.
      - Equal = correct pw.
    - Hashing calculation takes time for cracking…

Exploitation

- Password Cracking
  - What is Hashing?
    - Pa in
      m ature.
      B ed
      ri word
      h
    - H used
      to d in a
      h .
    - W e
      ve is
      h
      h ney
      m s not
      au

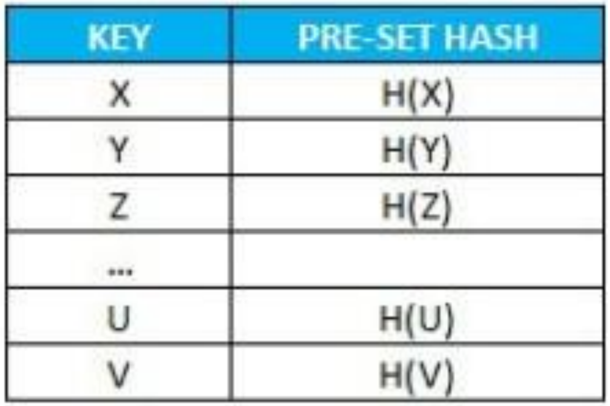

Input
Digest
Fox
cryptographic hash function
DFCD 3454 BBEA 788A 751A 696C 24D9 7009 CA99 2D17
The red fox jumps over the blue dog
cryptographic hash function
0086 46BB FB7D CBE2 823C ACC7 6CD1 90B1 EE6E 3ABC
The red fox jumps ouer the blue dog
cryptographic hash function
8FD8 7558 7851 4F32 D1C6 76B1 79A9 0DA4 AEFE 4819
The red fox jumps oevr the blue dog
cryptographic hash function
FCD3 7FDB 5AF2 C6FF 915F D401 C0A9 7D9A 46AF FB45
The red fox jumps oer the blue dog
cryptographic hash function
8ACA D682 D588 4C75 4BF4 1799 7D88 BCF8 92B9 6A6C

Exploitation

- Password Cracking
  - What are Rainbow Tables
    - Rainbow tables compute every possible combination of characters prior to capturing a password.
    - Once all the passwords have been generated, the attacker can capture the password hash and compare it to the results.

Exploitation

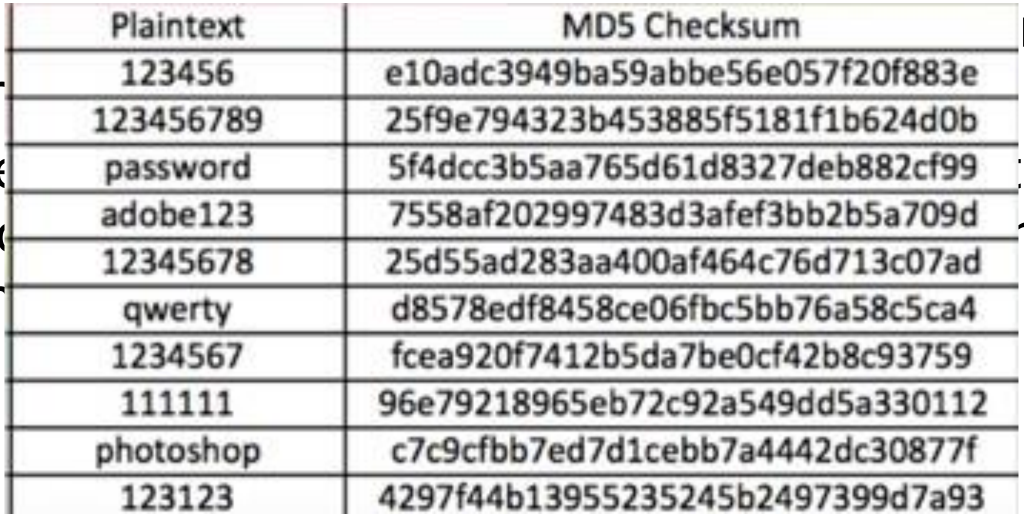
- Password Cracking
  - Rainbow Table:
    - Uses pre-created table with hashes:
    - Compare found hash with table.
      - H(x)?, H(Y)?, H(Z)=OK
    - Key is found.
      - Key must be Z



KEY	PRE-SET HASH
X	H(X)
Y	H(Y)
Z	H(Z)
...	
U	H(U)
V	H(V)

Exploitation

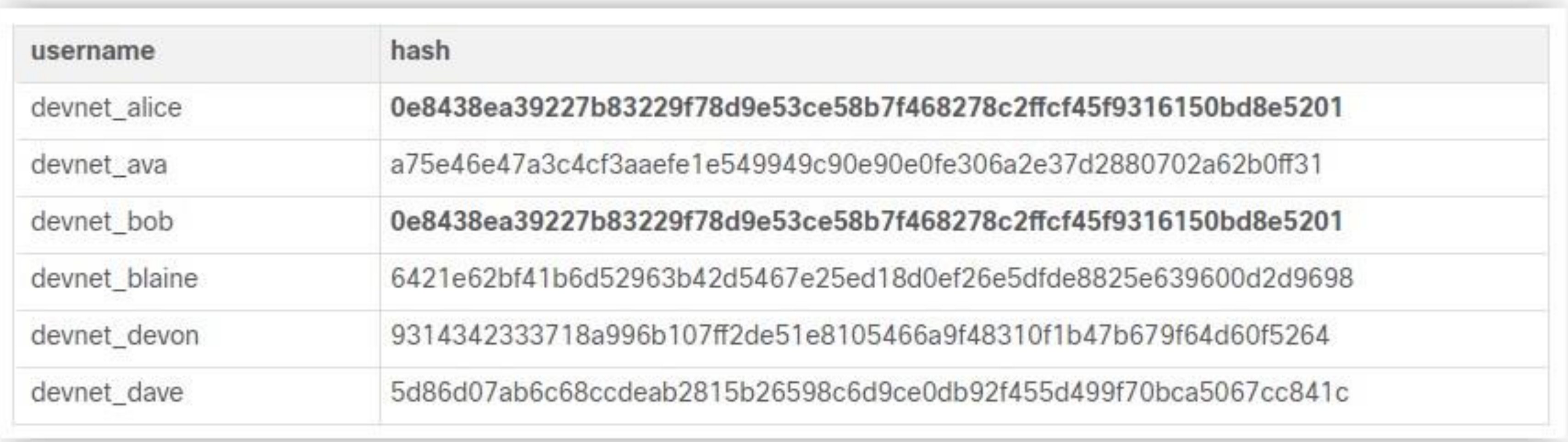
- Password Cracking
  - What are Rainbow Tables
    - Rain mbination of char
    - Once ed, the attac nd compare it to th



Plaintext	MD5 Checksum
123456	e10adc3949ba59abbe56e057f20f883e
123456789	25f9e794323b453885f5181f1b624d0b
password	5f4dcc3b5aa765d61d8327deb882cf99
adobe123	7558af202997483d3afef3bb2b5a709d
12345678	25d55ad283aa400af464c76d713c07ad
qwerty	d8578edf8458ce06fbc5bb76a58c5ca4
1234567	fcea920f7412b5da7be0cf42b8c93759
111111	96e79218965eb72c92a549dd5a330112
photoshop	c7c9cfbb7ed7d1cebb7a4442dc30877f
123123	4297f44b13955235245b2497399d7a93

Exploitation

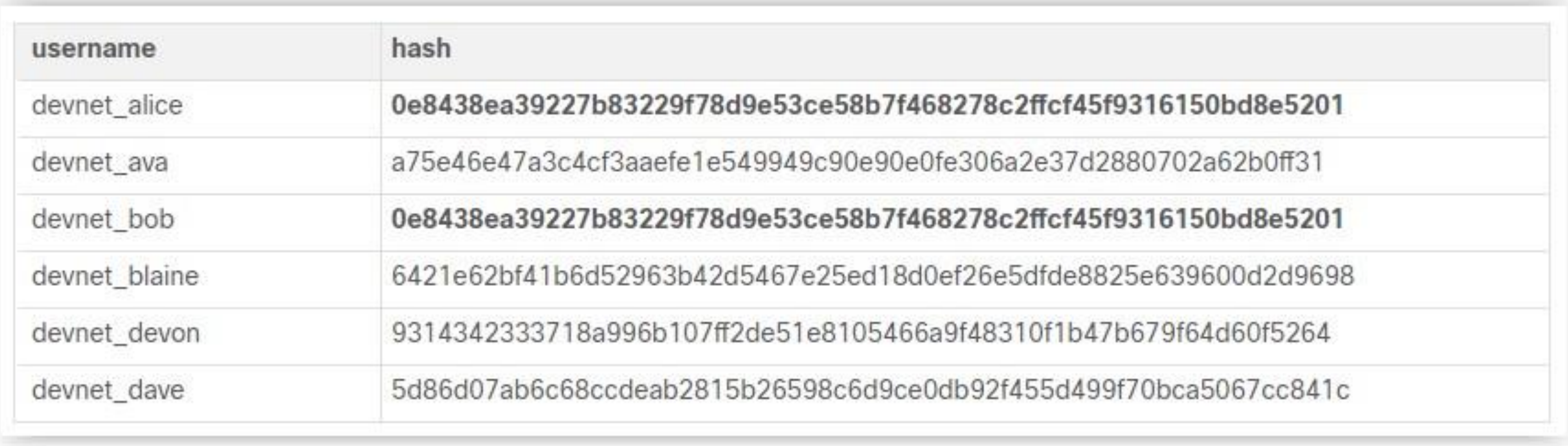
- Password Cracking
  - Hashing issues :
    - Rainbow tables
    - And…



username	hash
devnet_alice	**0e8438ea39227b83229f78d9e53ce58b7f468278c2ffcf45f9316150bd8e5201**
devnet_ava	a75e46e47a3c4cf3aaefe1e549949c90e90e0fe306a2e37d2880702a62b0ff31
devnet_bob	**0e8438ea39227b83229f78d9e53ce58b7f468278c2ffcf45f9316150bd8e5201**
devnet_blaine	6421e62bf41b6d52963b42d5467e25ed18d0ef26e5dfde8825e639600d2d9698
devnet_devon	9314342333718a996b107ff2de51e8105466a9f48310f1b47b679f64d60f5264
devnet_dave	5d86d07ab6c68ccdeab2815b26598c6d9ce0db92f455d499f70bca5067cc841c

Exploitation

- Password Cracking
  - Hashing issues :
    - Rainbow tables
    - Identify identical passwords (devnet_password1)



username	hash
devnet_alice	**0e8438ea39227b83229f78d9e53ce58b7f468278c2ffcf45f9316150bd8e5201**
devnet_ava	a75e46e47a3c4cf3aaefe1e549949c90e90e0fe306a2e37d2880702a62b0ff31
devnet_bob	**0e8438ea39227b83229f78d9e53ce58b7f468278c2ffcf45f9316150bd8e5201**
devnet_blaine	6421e62bf41b6d52963b42d5467e25ed18d0ef26e5dfde8825e639600d2d9698
devnet_devon	9314342333718a996b107ff2de51e8105466a9f48310f1b47b679f64d60f5264
devnet_dave	5d86d07ab6c68ccdeab2815b26598c6d9ce0db92f455d499f70bca5067cc841c

Exploitation

- Password Cracking
  - Hashing issues :
    - Rainbow tables
    - Identify identical passwords
  - Solution
    - “salting”

Exploitation

- Password Cracking
  - Hashing issues :
    - Rainbow tables
    - Identify identical passwords
  - Solution
    - “salting”
      - Salt = random string
      - Devnet_bob & devnet_allice both have pw devnet_password1
      - Salt for alice: salt706173776f726473616c74a
      - Salt for bob: salt706173776f726473616253b
      - Now calculate hash with password + salt (before or after pw)
      - Store hash & salt

Exploitation

Hashed and salted password examples

- User: devnet_alice
  - Password: devnetpassword1
  - Salt: salt706173776f726473616c74a
  - Salted input: devnetpassword1salt706173776f726473616c74a
  - Hash (SHA-256): cefee7f060ed49766d75bd4ca2fd119d7fcabe795b9425f4fa9d7115f355ab8c

- User: devnet_bob
  - Password: devnetpassword1
  - Salt: salt706173776f726473616253b
  - Salted input: devnetpassword1salt706173776f726473616253b
  - Hash (SHA-256): 41fffe05d7aca370abaff6762443d9326ce22107783b8ff5bb0cf576020fc1d5

Exploitation

- Password Cracking
  - Offline Attacks
    - Gather information from the computer and then analyse at home by any of the other cracking techniques

Exploitation

- Password Cracking
  - Nontechnical Attacks
    - Social engineering
    - Dumpster crawling
    - …

“Passwords are like underwear :
don’t let people see it, change it very often,
and you shouldn’t share it with strangers.”

Exploitation

- Methodical Approach
  - Password Cracking
  - Privilege Escalation
  - Executing Applications
  - Hiding files, covering tracks and concealing evidence
  - Expand the attack…

Exploitation

Privilege Escalation

Exploitation

- Methodical Approach
  - Privilege Escalation
    - Working from low-level privileges to high-level privileges
    - Horizontal Privilege Escalation
      - An attacker attempts to take over the rights and privileges of another user who has the same privileges as the current account.
    - Vertical Privilege Escalation
      - The attacker gains access to an account and then tries to elevate the privileges of the account. It is also possible to carry out a vertical escalation by compromising (= using the imperfections within the OS) an account and then trying to gain access to a higher-privileged account.
    - Utilities exist: Active Password changer, trinity rescue kit, ERD commander, Win RE, password resetter, ….

Exploitation

- Methodical Approach
  - Password Cracking
  - Privilege Escalation
  - Executing Applications
  - Hiding files, covering tracks and concealing evidence
  - Expand the attack…

Exploitation

Executing Applications

Exploitation

- Methodical Approach
  - Executing Applications
  - Backdoors
    - Applications of this type are designed to compromise the system in such a way as to allow later access to take place. An attacker can use these backdoors later to attack the system. Backdoors can come in the form of rootkits, Trojans, and similar types. They can even include software in the form of remote access Trojans (RATs).
  - Crackers
    - Any software that fits into this category is characterized by the ability to crack code or obtain passwords.
  - Keyloggers
    - Keyloggers are hardware or software devices used to gain information entered via the keyboard.

Exploitation

- Methodical Approach
  - Executing Applications
- Malware
  - This is any type of software designed to capture information, alter, or compromise the system.
    - » Virus: malicious code that must infect a host program to run.
    - » Worms: malicious code that is able to replicate itself without intervention.
    - » Trojans: programs completely written to look like a legitimate program, but actually carry malware within. The cracker usually uses popular software as bait and hides malware inside using special programs called wrappers.
    - » Hybrids: are malicious programs that can combine multiple functionalities in one program and also have been programmed not to be detected (using packaging and code obfuscation techniques). These kind of programs are even in many cases able to defend from antivirus systems.

Exploitation

- Methodical Approach
  - Password Cracking
  - Privilege Escalation
  - Executing Applications
  - Hiding files, covering tracks and concealing evidence
  - Expand the attack…

Exploitation

Hiding tracks

Exploitation

- Methodical Approach
  - Hiding files, covering tracks and concealing evidence
    - Prevent discovery
      - Disable Auditing
      - Eliminate Error Messages
      - Clean log files
      - …
    - Hide used files (file attributes)
    - Not important for ethical hacking

Exploitation

- Methodical Approach
  - Password Cracking
  - Privilege Escalation
  - Executing Applications
  - Hiding files, covering tracks and concealing evidence
  - Expand the attack…

Expand the attack…

Exploitation

- Methodical Approach
  - If possible expand the attack = new systems (methodology/PDCA cycle continues…)

Exploitation

- General hacking mechanisms
- Methodical Approach
  - Password Cracking
  - Privilege Escalation
  - Executing Applications
  - Hiding files, covering tracks and concealing evidence
  - Expand the attack…
- Frameworks
- Defense

Exploitation

Frameworks

Exploitation

- Frameworks (differences)
  - Reconnaisance
  - Scanning
  - Vulnerability scanning
  - Hacking/Exploitation

Can be all in 1 tool

- Examples:

❑ Metasploit

❑ Core Impact

❑ Immunity Canvas

- Usually with different versions: pro (pay), community (free) etc…

Exploitation

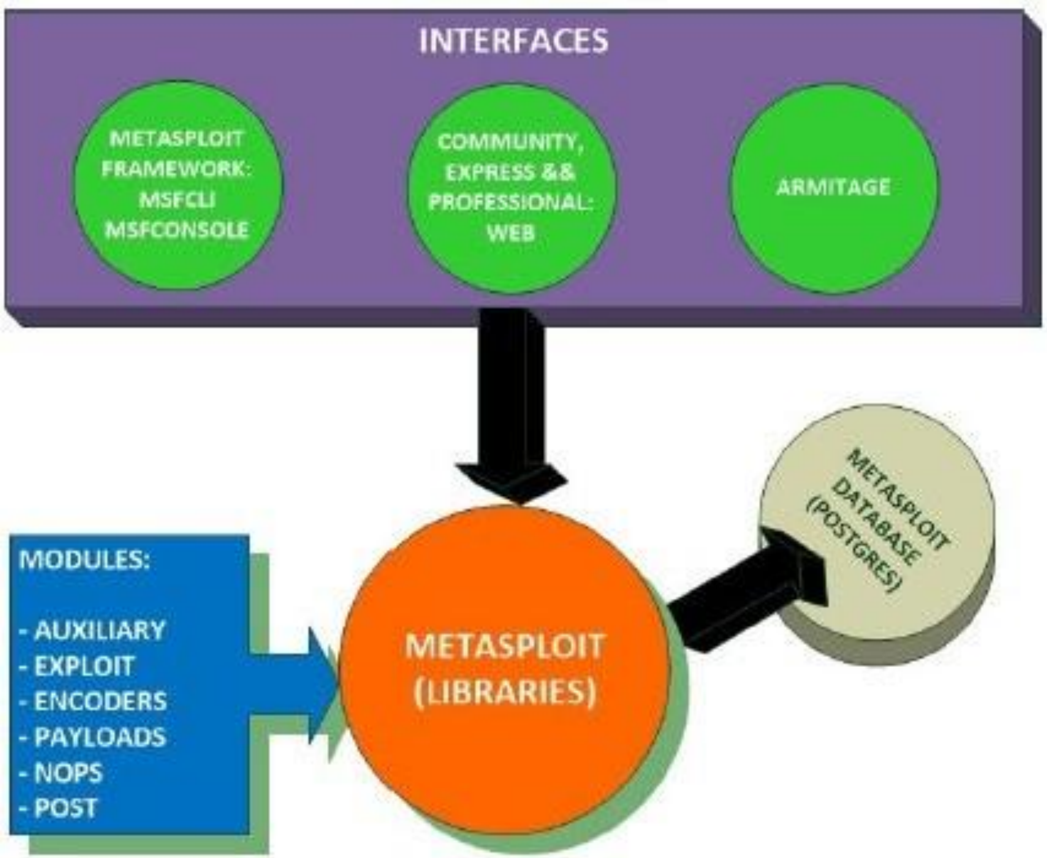
- Frameworks
  - Results
    - False Positive
      - Vulnerability found, but after investigation it’s not really a vulnerability.
    - False Negative
      - Vulnerability not found, but after investigation there’s a vulnerability.
    - True Positive
    - True Negative

Exploitation

- Framework: Metasploit
  - Rapid7
  - Commercial (Pay)
  - Part of KALI (framework)
  - Language = Ruby

Exploitation

- Framework: Metasploit
  - Metasploit Architecture


INTERFACES
METASPLOIT FRAMEWORK: MSFCLI MSFCONSOLE
COMMUNITY, EXPRESS && PROFESSIONAL: WEB
ARMITAGE
METASPLOIT DATABASE (POSTGRES)
MODULES:
- AUXILIARY
- EXPLOIT
- ENCODERS
- PAYLOADS
- NOPS
- POST
METASPLOIT (LIBRARIES)

METASPLOIT ARCHITECTURE
SOURCE: Offensive Security (2013), *Metasploit Unleashed.*

Exploitation

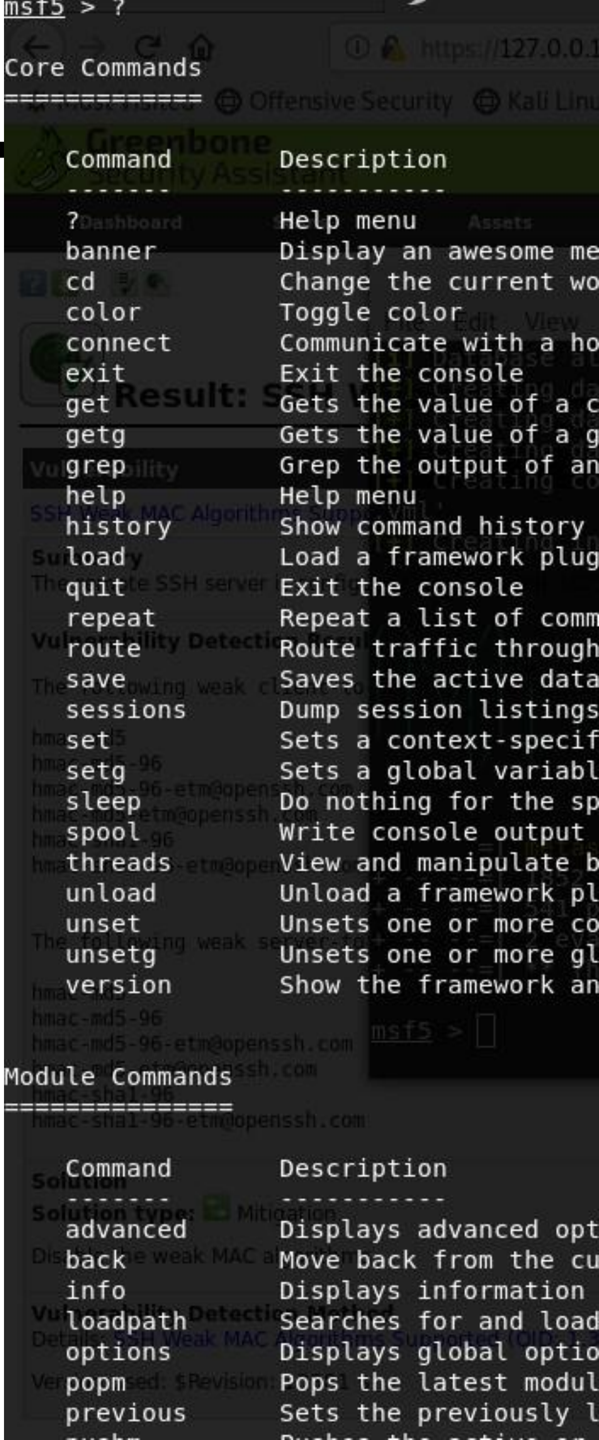
- Framework: Metasploit
  - Metasploit Architecture
    - Libraries
      - Manage basic functionality
      - Provide the different functions (APIs) for the interfaces
      - Interact with the supported protocols
    - Interfaces
      - Msfcli (scripting) (non existent, -x option in msfconsole?)
      - Msfconsole
      - Web
      - Armitage (GUI)
    - File System
      - Directories: data, lib, modules, plugins, scripts, tools
      - Inside: /opt/metasploit or /usr/share/metasploit-framework
    - Modules

Exploitation

- Framework: Metasploit
  - Metasploit Architecture
  - Modules
    - 1. Auxiliary
      - » Functionality = scanning ports, logging
    - 2. Encoders
      - » Encoding/decoding payloads
    - 3. Exploits
      - » Target specific system vulnerabilities for access. Use payloads = code to execute remotely.
    - 4. Nops
      - » Complex operations that are used within MSF to ensure the proper execution of a payload or provide stability to it.
    - 5. Payloads
      - » Payloads are programs that run remotely on a victim host after an exploit is successfull
    - 6. Post
      - » Used to gain greater access, maintain it up or get further information from a victim host, after this has been compromised.

Exploitation

- Framework: Metasploit
  - Metasploit Architecture
    - MSF Console
      - Shell type environment
      - Commands can be executed
      - Type ? to get all commands:
        - » Core
        - » Module
        - » Job
        - » Resource Script
        - » Database Backend
        - » Credentials Backend
        - » Developer



```
msf5 > ?

Core Commands
=============

    Command       Description
    -------       -----------
    ?             Help menu
    banner        Display an awesome me
    cd            Change the current wo
    color         Toggle color
    connect       Communicate with a ho
    exit          Exit the console
    get           Gets the value of a c
    getg          Gets the value of a g
    grep          Grep the output of an
    help          Help menu
    history       Show command history
    load          Load a framework plug
    quit          Exit the console
    repeat        Repeat a list of comm
    route         Route traffic through
    save          Saves the active data
    sessions      Dump session listings
    set           Sets a context-specif
    setg          Sets a global variabl
    sleep         Do nothing for the sp
    spool         Write console output
    threads       View and manipulate b
    unload        Unload a framework pl
    unset         Unsets one or more co
    unsetg        Unsets one or more gl
    version       Show the framework an


Module Commands
===============

    Command       Description
    -------       -----------
    advanced      Displays advanced opt
    back          Move back from the cu
    info          Displays information
    loadpath      Searches for and load
    options       Displays global optio
    popm          Pops the latest modul
    previous      Sets the previously l
    pushm         Pushes the active or
```

Exploitation

- Framework: Metasploit
  - Metasploit Architecture
  - Workspaces
    - Store information collected during audits
    - Saved in internal Postgres database
    - Workspace –a NameOfWorkspace = create
    - Workspace NameOfWorkspace = use
    - Workspace = all workspaces

Exploitation

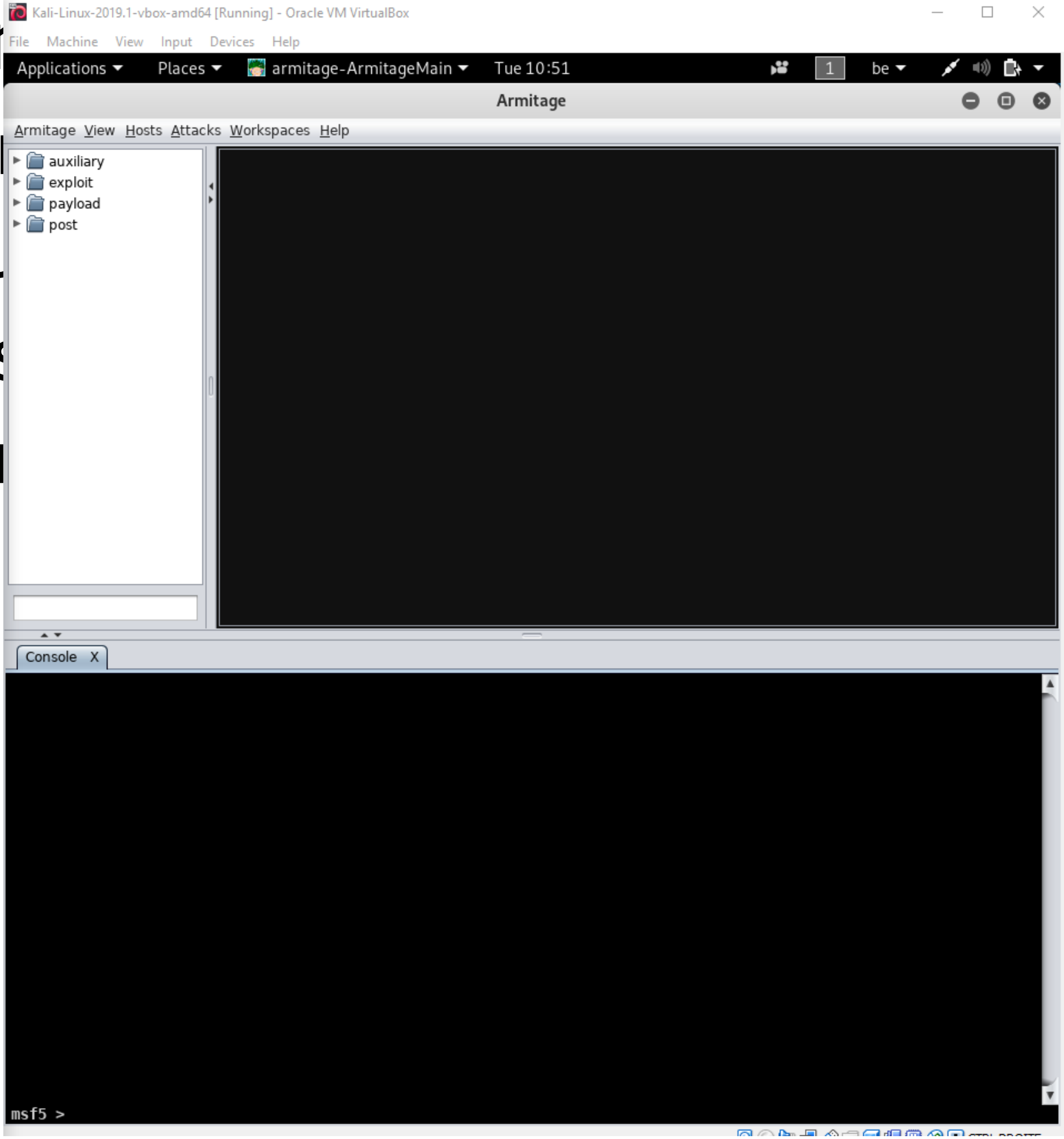
- Framework: Metasploit
  - Metasploit Architecture
    - Community Edition
    - Has web browser on port 3790
    - Doesn’t work by default?

Exploitation

- Framework: Metasploit
  - Armitage
  - GUI to Metasploit
  - Installed by default in KALI
  - Run: Armitage & (+ few boxes to click)

Exploitation

- Fra
  - Ar
  - GU
  - Ins
  - Ru o click)


Kali-Linux-2019.1-vbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places armitage-ArmitageMain Tue 10:51 1 be
Armitage
Armitage View Hosts Attacks Workspaces Help
auxiliary
exploit
payload
post
Console X

```
msf5 >
```

Exploitation

- General hacking mechanisms
- Methodical Approach
  - Password Cracking
  - Privilege Escalation
  - Executing Applications
  - Hiding files, covering tracks and concealing evidence
  - Expand the attack…
- Frameworks
- Defense

Exploitation

Defense

Exploitation

- DEFENSE:
  - Create a sufficient password security policy (length, use of special chars, expiration, blocking, …)
  - Enable auditing on OS-level of end-user devices, servers, communication equipment.
  - Use logging and event-monitoring software.
  - Restrict Administrator/Root account = only logon locally

Exploitation

- DEFENSE:
  - Use port security and admission control (NAC) on networking devices so that only authorized users can connect to the network.
  - Replace insecure protocols that send information in plain text as HTTP, SMTP, TELNET, FTP, with their secure counterparts which use digital certificates and encryption for transmission: HTTPS, SMTP (higher version), SSL, SSH, SFTP, etc.
  - Set the switches to detect the sending of free and unauthorized ARP and other known attacks and react to port violation taking appropriate actions and reporting the event.
  - Implement secure authentication protocols in wireless equipment and isolate wireless segments from other internal subnets using intelligent next generation firewalls.

Exploitation

- DEFENSE:
  - Configure intelligent next generation firewalls and other network devices to block attacks.
  - Use network and security management software for threat detection, vulnerability assessment and automatic response to events69.
  - Design and implement an Information Security Policy based on the ISO 27000 standard.
  - Implement awareness campaigns about good practices on information security for the end-users.
  - Train staff from the IT and related departments about information security and specialized topics such as ethical hacking, computer forensics and defense mechanisms.
  - Define profiles for IT personnel and establish which international certifications on information security your functionaries must obtain according to their position.