Cybersecurity

Penetration Test Report Template

MegaCorpOne

Penetration Test Report

[Bot Intruder Testing], LLC

Confidentiality Statement

This document contains confidential and privileged information from MegaCorpOne Inc. (henceforth known as MegaCorpOne). The information contained in this document is confidential and may constitute inside or non-public information under international, federal, or state laws. Unauthorized forwarding, printing, copying, distribution, or use of such information is strictly prohibited and may be unlawful. If you are not the intended recipient, be aware that any disclosure, copying, or distribution of this document or its parts is prohibited.

Table of Contents

Contact Information

Company Name	Bot Intruder Testing, LLC
Contact Name	Adam Gonzalez
Contact Title	Penetration Tester
Contact Phone	555.224.2411
Contact Email	AdamGonzalez@BIT.com

Document History

Version	Date	Author(s)	Comments
001	04/21/2003	Adam Gonzalez	

Introduction

In accordance with MegaCorpOne's policies, Bot Intruder Testing, LLC (henceforth known as [BIT) conducts external and internal penetration tests of its networks and systems throughout the year. The purpose of this engagement was to assess the networks' and systems' security and identify potential security flaws by utilizing industry-accepted testing methodology and best practices. The project was conducted on a number of systems on MegaCorpOne's network segments by [BIT] during Apr of 2023.

For the testing, BIT focused on the following:

- Attempting to determine what system-level vulnerabilities could be discovered and exploited with no prior knowledge of the environment or notification to administrators.
- Attempting to exploit vulnerabilities found and access confidential information that may be stored on systems.
- Documenting and reporting on all findings.

All tests took into consideration the actual business processes implemented by the systems and their potential threats; therefore, the results of this assessment reflect a realistic picture of the actual exposure levels to online hackers. This document contains the results of that assessment.

Assessment Objective

The primary goal of this assessment was to provide an analysis of security flaws present in MegaCorpOne's web applications, networks, and systems. This assessment was conducted to identify exploitable vulnerabilities and provide actionable recommendations on how to remediate the vulnerabilities to provide a greater level of security for the environment.

[BIT] used its proven vulnerability testing methodology to assess all relevant web applications, networks, and systems in scope.

MegaCorpOne has outlined the following objectives:

Table 1: Defined Objectives

Objective
Find and exfiltrate any sensitive information within the domain.
Escalate privileges to domain administrator.
Compromise at least two machines.

Penetration Testing Methodology

Reconnaissance

[BIT] begins assessments by checking for any passive (open source) data that may assist the assessors with their tasks. If internal, the assessment team will perform active recon using tools such as Nmap and Bloodhound.

Identification of Vulnerabilities and Services

[BIT] uses custom, private, and public tools such as Metasploit, hashcat, and Nmap to gain perspective of the network security from a hacker's point of view. These methods provide MegaCorpOne with an understanding of the risks that threaten its information, and also the strengths and weaknesses of the current controls protecting those systems. The results were achieved by mapping the network architecture, identifying hosts and services, enumerating network and system-level vulnerabilities, attempting to discover unexpected hosts within the environment, and eliminating false positives that might have arisen from scanning.

Vulnerability Exploitation

[BIT]'s normal process is to both manually test each identified vulnerability and use automated tools to exploit these issues. Exploitation of a vulnerability is defined as any action we perform that gives us unauthorized access to the system or the sensitive data.

Reporting

Once exploitation is completed and the assessors have completed their objectives, or have done everything possible within the allotted time, the assessment team writes the report, which is the final deliverable to the customer.

Scope

Prior to any assessment activities, MegaCorpOne and the assessment team will identify targeted systems with a defined range or list of network IP addresses. The assessment team will work directly with the MegaCorpOne POC to determine which network ranges are in-scope for the scheduled assessment.

It is MegaCorpOne's responsibility to ensure that IP addresses identified as in-scope are actually controlled by MegaCorpOne and are hosted in MegaCorpOne-owned facilities (i.e., are not hosted by an external organization). In-scope and excluded IP addresses and ranges are listed below.

IP Address/URL	Description
172.16.117.0/16 MCO.local *.Megacorpone.com	MegaCorpOne internal domain, range and public website

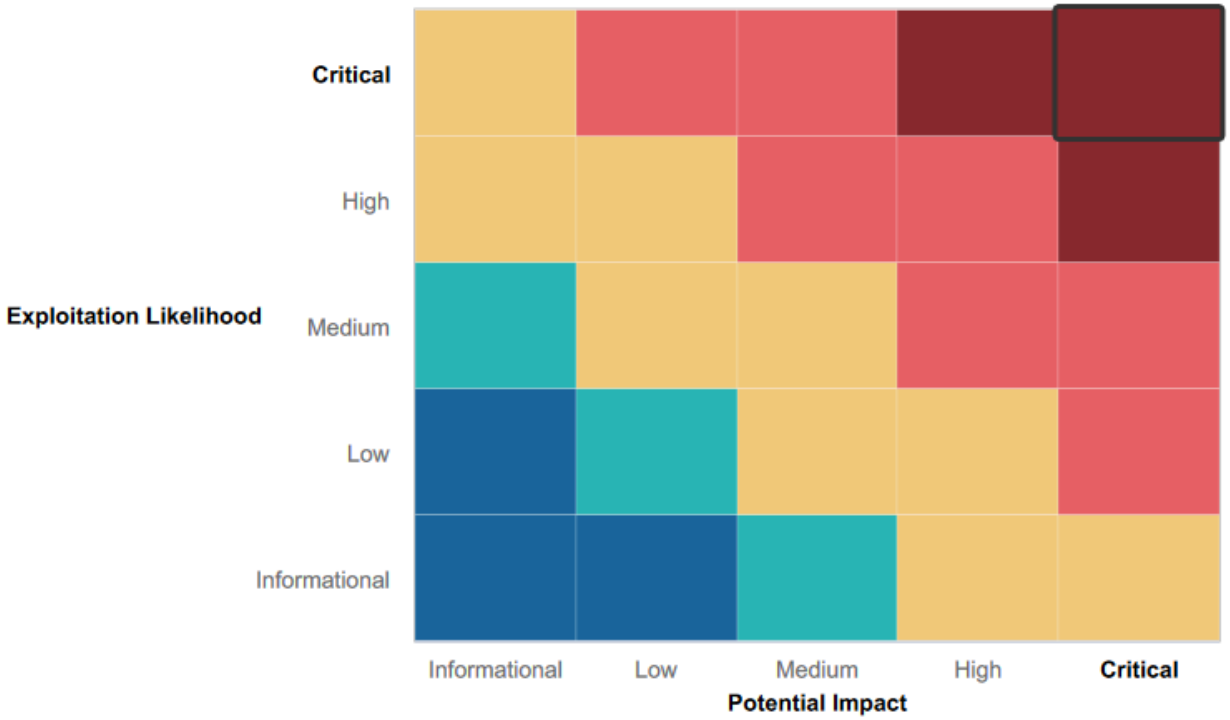
Executive Summary of Findings

Grading Methodology

Each finding was classified according to its severity, reflecting the risk each such vulnerability may pose to the business processes implemented by the application, based on the following criteria:

Critical: Immediate threat to key business processes.
High: Indirect threat to key business processes/threat to secondary business processes.
Medium: Indirect or partial threat to business processes.
Low: No direct threat exists; vulnerability may be leveraged with other vulnerabilities.
Informational: No threat; however, it is data that may be used in a future attack.

As the following grid shows, each threat is assessed in terms of both its potential impact on the business and the likelihood of exploitation:



Summary of Strengths

While the assessment team was successful in finding several vulnerabilities, the team also recognized several strengths within MegaCorpOne's environment. These positives highlight the effective countermeasures and defenses that successfully prevented, detected, or denied an attack technique or tactic from occurring.

- Other ports on machine not weak to attacks
- 997 tcp ports closed
- backdoor on web app failed
- mpf exploit failed
- failed vbseo_proc_deutf

- strong http defense

Summary of Weaknesses

[BIT] successfully found several critical vulnerabilities that should be immediately addressed in order to prevent an adversary from compromising the network. These findings are not specific to a software version but are more general and systemic vulnerabilities.

- Weak and Easy to guess passwords
- admin.txt password file
- WMI exploit
- LLMNR exploit
- Zenmap port scanning
- smbclient not restricted
- No firewalls
- No monitor logging
- task scheduler enabled

Executive Summary

BIT is running a pentesting assessment on companies site megacorpone.com. BIT has found vulnerabilities on your Web application, Linux systems, and Windows systems. This summary is to give you a better outlook on how all these systems are linked together to create a stronger system in the future. BIT started off with port scanning and noticed that port 21 was open. It fell to a backdoor attack and using that BIT was able to get credentials. Then using those we got admin access and privilege escalation with a password file. After privilege escalation BIT got all the password hashes and cracked them giving access to more accounts. Maintaining persistence BIT made a fake system program to allow us to go back to the system whenever we wanted to. Noticing that the accounts were on windows servers we used credentials to login. Then BIT wanted to poke holes in multiple different areas so we ran LLMNR to get some feedback and we got back an account that would be cracked and used. Then using msfvemon a payload was able to be uploaded and used to access and migrate to different systems and use the task scheduler to have payload upload at the time of the day we wanted it to. Using the kiwi was important to getting the mscashv2 hashes to crack. Then putting in those credentials we could go from Windows10 to WINDC01 successfully achieving lateral movement and getting to the NTLM hashes to crack domain controller user credentials. BIT uses all sorts of different angles to attack not just one but all three and they are connected very nicely. There are just some bumps in the road that could potentially lead to data loss, user exploitation, and web exploitation. This assessment was used to determine what those were and highlight them to you with ideas to fix it and provide mitigation strategies to create a well rounded system.

Summary Vulnerability Overview

Vulnerability	Severity
Weak password on public web application	Critical
exploit/unix/ftp/vsftpd_234_backdoor	critical
/usr/share/exploitdb/exploits/unix/remote/49757.py script	Critical
admin.txt file with msfadmin password in it leading to cracked passwords with john	critical
exploit/auxiliary/scanner/smb/smb_login	critical
LLMNR spoofing	High
auxiliary/scanner/smb/impacket/wmiexec	Medium
msfvemon payload with exploit/multi/handler and scanner/smb/impacket/wmiexec	Critical
exploit/windows/local/persistence_service	High
Created scheduled task for payload upload	Critical
exploit/windows/smb/psexec using migrate command	Critical
exploit/windows/smb/psexec and exploit/windows/local/wmi	Critical
Using meterpreter shell with scync_ntlm to get NTLM hashes	Critical
In shell adding more ports for persistence via port 10022	High
Zenmap for port scanning	Low

The following summary tables represent an overview of the assessment findings for this penetration test:

Scan Type	Total
Hosts	151
Ports	23

Exploitation Risk	Total
Critical	10
High	3
Medium	1
Low	1

Vulnerability Findings

Weak Password on Public Web Application

Risk Rating: Critical

Description:
The site **vpn.megacorpone.com** is used to host the Cisco AnyConnect configuration file for MegaCorpOne. This site is secured with basic authentication but is susceptible to a dictionary attack. [BIT] was able to use a username gathered from OSINT in combination with a wordlist in order to guess the user's password and access the configuration file.

Affected Hosts: vpn.megacorpone.com

Remediation:

- Set up two-factor authentication instead of basic authentication to prevent dictionary attacks from being successful.
- Require a strong password complexity that requires passwords to be over 12 characters long, upper+lower case, & include a special character.
- Reset the user **thudson**'s password.

Zenmap Port Scanning

Risk Rating: Low

Description:
Using zenmap for website vpn.megacorpone.com we were able to find that port 21 was open. This port is susceptible to a backdoor attack which then the intruder can upload malicious code.

Affected Hosts: vpn.megacorpone.com

Remediation:

- A firewall that will be defensive against scanning
- Enable administrators to let certain ip addresses access the port and everyone else is blacklisted and off limits.

Backdoor Attack

Risk Rating: Critical

Description:
After using zenmap to figure out that port 21 was vulnerable to a backdoor attack we quickly made progress in uploading a /usr/share/exploits/unix/remote/49757.py. By passing this exploit to the ip address of 172.17.0.4 we were able to get a shell to further start digging and further escalate the attack.

Affected Hosts: vpn.megacorpone.com (Linux OS)

Remediation:

- Get an antivirus and malware protection software to not allow backdoor scripts to be uploaded.
- strong password policy.
- keep systems updated.

Msfexpolit Exploit

Risk Rating: Critical

Description:
Opening up further testing on vpn.megacorpone.com we decided to use msfexploit. After noticing that it was weak to ftp attacks we decided to hit it from a different angle. Using exploit/unix/ftp/vsftpd_234_backdoor and ip address 172.22.117.150, We succeeded in running the exploit and were able to get a response back.

Affected Hosts: vpn.megacorpone.com (Linux OS)

Remediation:

- Use SFTP it allows you to do the same thing FTP does but with security added.
- Place a gateway for FTP so it only allows what you want to see enter your network.

Privilege Escalation

Risk Rating: Critical

Description:
While in vpn.megacorpone.com as a daemon user we were looking for ways to get information to privilege escalate. While looking through this user we were able to find an admin.txt file that had “msfadmins” password in it. Using this user's name we were able to ssh into msfadmin and privilege escalate using the password “cybersecurity”.

Affected Hosts: vpn.megacorpone.com

Remediation:

- Having passwords in .txt files in the system creates a vulnerability for privilege escalation.
- Use stronger passwords

Password Cracking

Risk Rating: Critical

Description:
After getting into the host “msfadmin” with the password “cybersecurity”. We were able to go to /etc/shadow and discover all the password hashes. Copying them over to my kali terminal I was able to apply john to it and get them cracked giving me access to multiple different accounts inside of megacorpone.com.

Affected Hosts: vpn.megacorpone.com (Linux OS)

Remediation:

- Use stronger passwords to prevent root privileges

Persistence

Risk Rating: High

Description:
Being in the system and was going to eventually get noticed I decided it was needed to make a fake machine account that would be “working” in the background but in reality it was go come back and maintain persistence. I also added a new port to support login traffic that wouldn’t get noticed as much. Doing this I was able to achieve persistence into the system and come back if I needed to.

Affected Hosts: vpn.megacorpone.com (Linux OS)

Remediation:

- Limit access to devices
- Strong monitoring systems
- Keep eyes villagate on new accounts being made without permission

Password Spray

Risk Rating: Critical

Description:
Using the information that we got earlier noticing that the accounts at Megacorp were using windows machines we acted accordingly and used a password spraying exploit. The credentials user “tstark” and “Password!” is what we used to get into the system. It was successfully able to login into 172.22.117.20 ip address and run as admin.

Affected Hosts: vpn.megacorpone.com (Windows OS)

Remediation:

- Use strong passwords.
- Strong monitoring to see if certain ips look suspicious.
- Prevent scanning with firewalls and antivirus.

LLMNR Spoofing

Risk Rating: High

Description:
While having that one user we decided to use LLMNR spoofing to see if we could get some feedback. It was apparent that we were able to get the password hash callback. When we put them in john we got user “pparker” and password “Spring2021”.

Affected Hosts: vpn.megacorpone.com (Windows OS)

Remediation:

- Disable LLMNR
- Use strong passwords

WMI Attack

Risk Rating: **Medium**

Description:
Now that we had two users credentials we were going to leverage them and use “tstark” user to get some valuable information about the system. Using WMI, BIT was able to get into the system and run commands to find information that could be helpful to use later on in this pentesting.

Affected Hosts: vpn.megacorpone.com (Windows OS)

Remediation:

- Disable WMI

MSFvenom Payload Attack

Risk Rating: **Critical**

Description:
BIT creating a custom payload to run against your system with msfvemon. Using smbclient and getting into megacorpone BIT was able to upload the shell.exe file. The exploit multi/handler was also used to ensure that it was listening to our commands in the background. Back WMI we set the custom payload to the COMMAND prompt and we ran it getting a meterpreter session and got a response back.

Affected Hosts: vpn.megacorpone.com (Windows OS)

Remediation:

- Use firewalls to not allow smb across the internet
- Look in monitoring for .exe files and high usage rates for certain files.

Privilege Escalation

Risk Rating: **High**

Description:
Now that BIT had access to the machine via the payload meterpreter session we used it for privilege escalation. Using the windows/local/persistence_service BIT was able to act like we were a service but BIT was running malicious payloads to escalate. Then BIT would change the name of the payload to make it seem like a normal process.

Affected Hosts: vpn.megacorpone.com (Windows OS)

Remediation:

- Password policy so they can’t get into the system at first
- Preventing data execution
- Strong monitoring for logins and attacks

Persistence

Risk Rating: Critical

Description:
The site megacorpone.com is prone to task scheduler attacks when a malicious actor has gotten into the system. BIT has demonstrated how dangerous it could be by running a backdoor payload to run everyday at midnight. Using this technique could be deadly for getting daily information or using it to upload better attacks.

Affected Hosts: vpn.megacorpone.com (Windows OS)

Remediation:

- Strong monitoring to see what is in task scheduler
- Disable service if not in use

CredDumping

Risk Rating: Critical

Description:
BIT used kiwi in metasploit to dump credentials that are on the WIN10 machine. BIT used to smb/psexec exploit and ran it then loading kiwi we dumped the lsadump::cache and out came MsCacheV2 hashes. Copying them and using John BIT was able to get users usernames and passwords allowing us to login as more users on megacorpone.com.

Affected Hosts: vpn.megacorpone.com (Windows OS)

Remediation:

- Monitor service and databases regularly to prevent attacks
- Watch for command line arguments used for credential dumping
- Disable or restrict NTLM

Lateral Movement

Risk Rating: Critical

Description:
Having a windows shell opened and running in the background BIT then navigated to WMI putting in the credentials needed and successfully launching an exploit from WIndows10 to WINDC01 putting our options of attack more with variety. Successfully achieving lateral movement between multiple machines.

Affected Hosts: vpn.megacorpone.com (Windows OS)

Remediation:

- Implement threat and advanced behavior monitoring
- Require strong passwords and changed frequently

- Applications allow listing so if someone logs into an application that is not allowed it gets logged.

Credential Access

Risk Rating: **Critical**

Description:
BIT now has access to the domain controller which means that BIT has access to the NTLM hashes inside. Using kiwi and performing dcsync_ntlm BIT was able to get all the hashes and copy them over to our kali machine and crack them with john giving us users.

Affected Hosts: vpn.megacorpone.com (Windows OS)

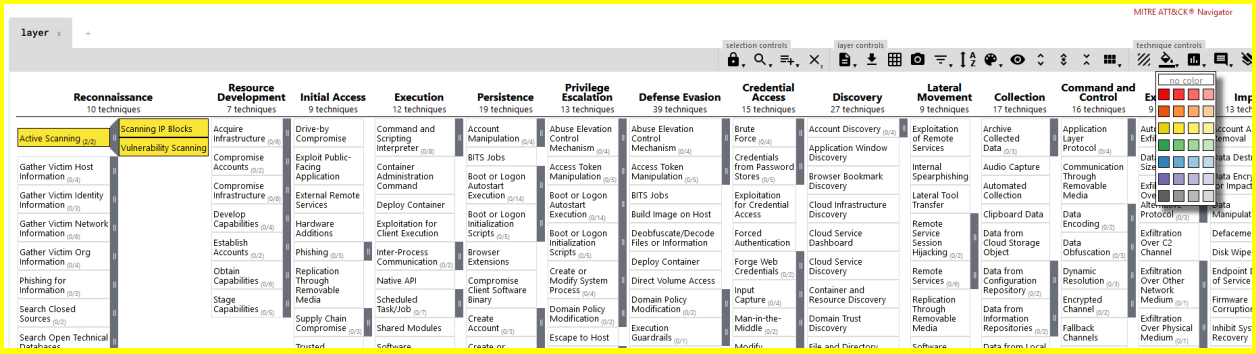
Remediation:

- Restrict NTLM
- Restrict access to domain controllers
- Monitoring domain controller access logs

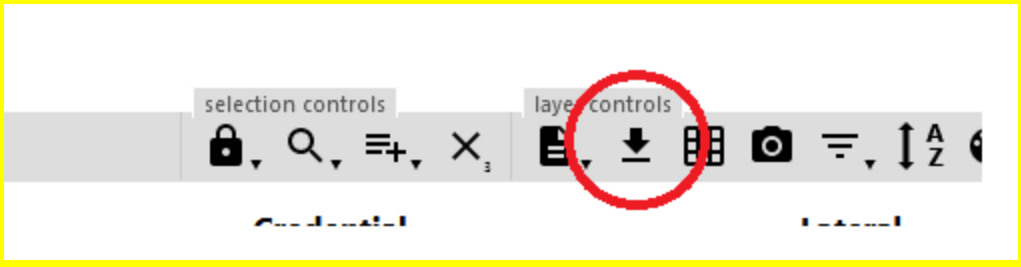
[List any other vulnerabilities you found here. Feel free to go into as much detail (including technical detail) as you want.]

MITRE ATT&CK Navigator Map

[Using the MITRE ATT&CK Navigator, build out a map showing what techniques you've used so far. To do so, on the MITRE ATT&CK Navigator page, click "Create New Layer," then "Enterprise," and select each technique that you've used. Change the color of each selected technique to highlight it in yellow if it was successful, or in red if it was unsuccessful, as the following image shows:



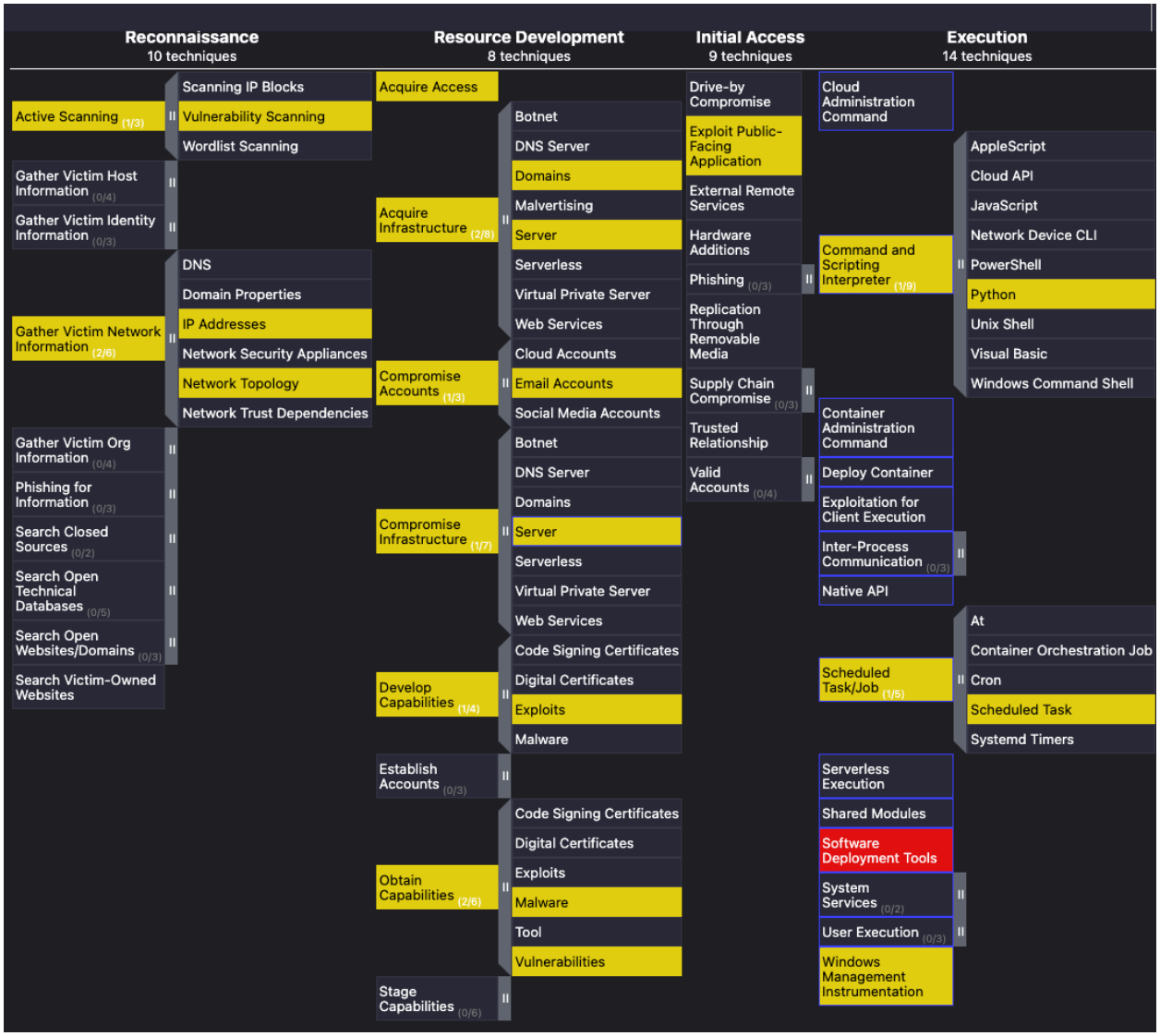
When you're done, be sure to download the chart as JSON by clicking the download icon, as the following image shows:

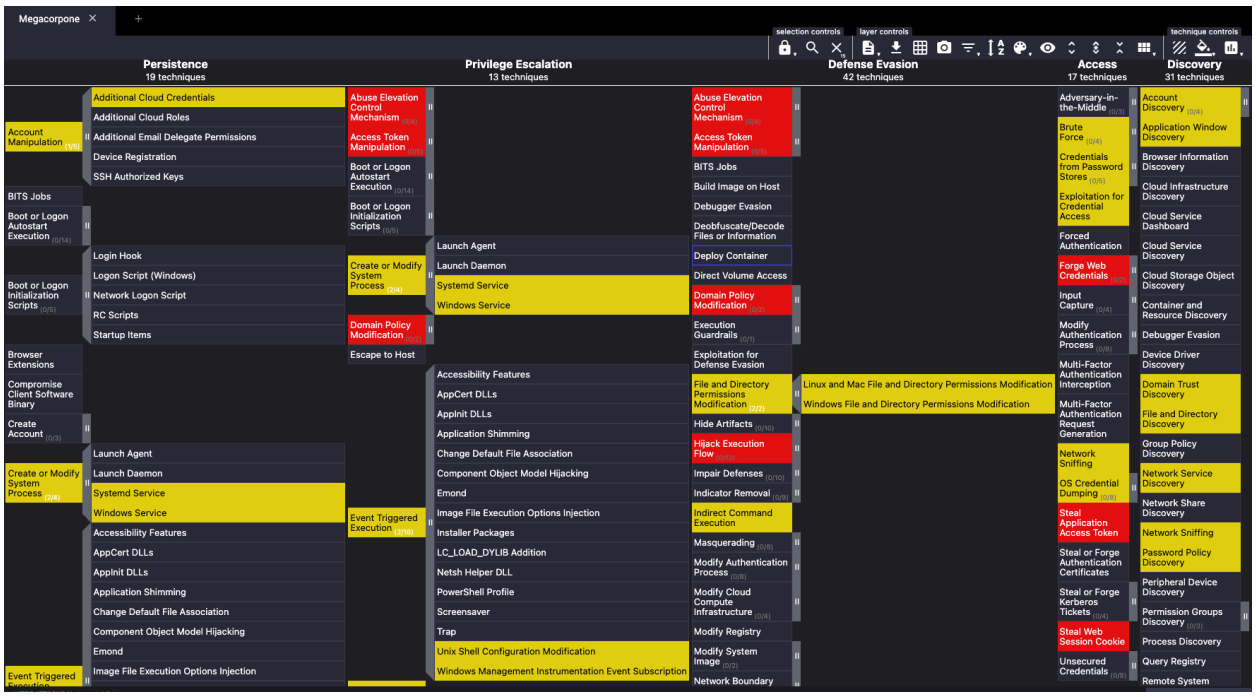


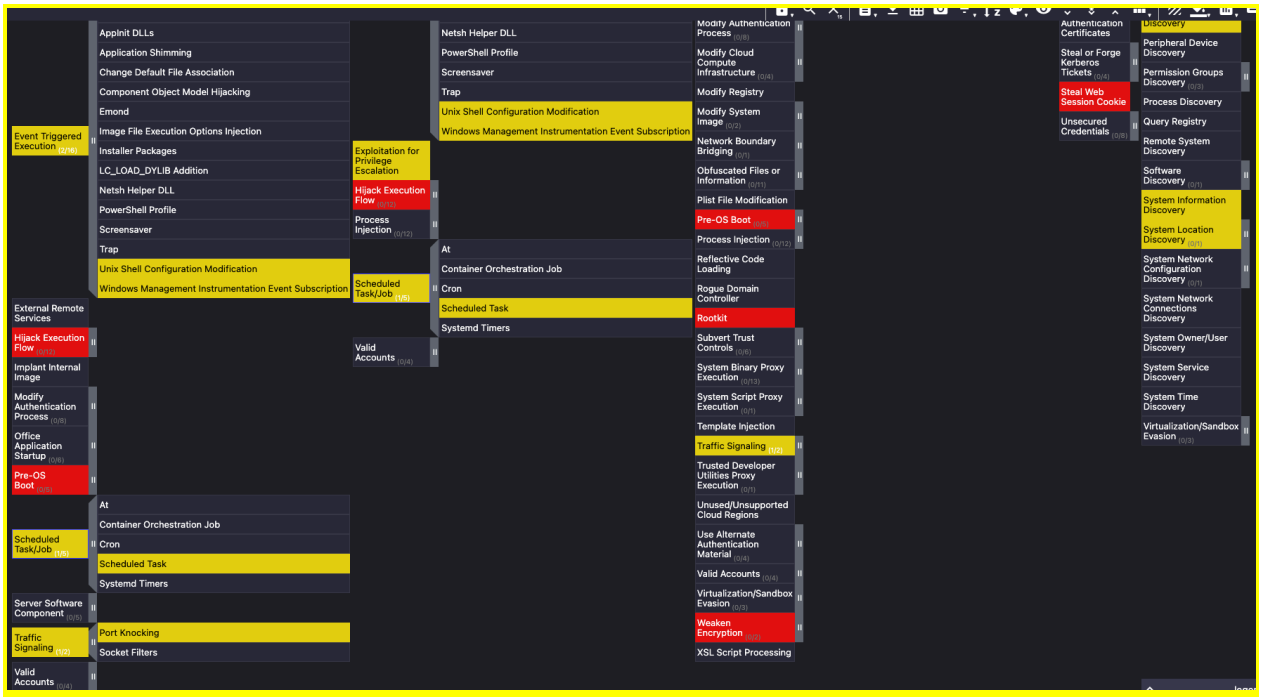
Remember, this report is not yet complete—we will finish it in the next module.

The following completed MITRE ATT&CK navigator map shows all of the techniques and tactics that [BIT] used throughout the assessment.

Legend:


Reconnaissance
10 techniques
Active Scanning (1/3)
Scanning IP Blocks
Vulnerability Scanning
Wordlist Scanning
Gather Victim Host Information (0/4)
Gather Victim Identity Information (0/3)
Gather Victim Network Information (2/6)
DNS
Domain Properties
IP Addresses
Network Security Appliances
Network Topology
Network Trust Dependencies
Gather Victim Org Information (0/4)
Phishing for Information (0/3)
Search Closed Sources (0/2)
Search Open Technical Databases (0/5)
Search Open Websites/Domains (0/3)
Search Victim-Owned Websites
Resource Development
8 techniques
Acquire Access
Acquire Infrastructure (2/8)
Botnet
DNS Server
Domains
Malvertising
Server
Serverless
Virtual Private Server
Web Services
Compromise Accounts (1/3)
Cloud Accounts
Email Accounts
Social Media Accounts
Compromise Infrastructure (1/7)
Botnet
DNS Server
Domains
Server
Serverless
Virtual Private Server
Web Services
Develop Capabilities (1/4)
Code Signing Certificates
Digital Certificates
Exploits
Malware
Establish Accounts (0/3)
Obtain Capabilities (2/6)
Code Signing Certificates
Digital Certificates
Exploits
Malware
Tool
Vulnerabilities
Stage Capabilities (0/6)
Initial Access
9 techniques
Drive-by Compromise
Exploit Public-Facing Application
External Remote Services
Hardware Additions
Phishing (0/3)
Replication Through Removable Media
Supply Chain Compromise (0/3)
Trusted Relationship
Valid Accounts (0/4)
Execution
14 techniques
Cloud Administration Command
Command and Scripting Interpreter (1/9)
AppleScript
Cloud API
JavaScript
Network Device CLI
PowerShell
Python
Unix Shell
Visual Basic
Windows Command Shell
Container Administration Command
Deploy Container
Exploitation for Client Execution
Inter-Process Communication (0/3)
Native API
Scheduled Task/Job (1/5)
At
Container Orchestration Job
Cron
Scheduled Task
Systemd Timers
Serverless Execution
Shared Modules
Software Deployment Tools
System Services (0/2)
User Execution (0/3)
Windows Management Instrumentation


Megacorpone
Persistence
19 techniques
Privilege Escalation
13 techniques
Defense Evasion
42 techniques
Access
17 techniques
Discovery
31 techniques
Account Manipulation (1/5)
Additional Cloud Credentials
Additional Cloud Roles
Additional Email Delegate Permissions
Device Registration
SSH Authorized Keys
BITS Jobs
Boot or Logon Autostart Execution (0/14)
Boot or Logon Initialization Scripts (0/5)
Login Hook
Logon Script (Windows)
Network Logon Script
RC Scripts
Startup Items
Browser Extensions
Compromise Client Software Binary
Create Account (0/3)
Create or Modify System Process (2/4)
Launch Agent
Launch Daemon
Systemd Service
Windows Service
Accessibility Features
AppCert DLLs
AppInit DLLs
Application Shimming
Change Default File Association
Component Object Model Hijacking
Emond
Image File Execution Options Injection
Event Triggered Execution (2/16)
Abuse Elevation Control Mechanism (0/4)
Access Token Manipulation (0/5)
Boot or Logon Autostart Execution (0/14)
Boot or Logon Initialization Scripts (0/5)
Create or Modify System Process (2/4)
Launch Agent
Launch Daemon
Systemd Service
Windows Service
Domain Policy Modification (0/2)
Escape to Host
Accessibility Features
AppCert DLLs
AppInit DLLs
Application Shimming
Change Default File Association
Component Object Model Hijacking
Emond
Image File Execution Options Injection
Installer Packages
LC_LOAD_DYLIB Addition
Netsh Helper DLL
PowerShell Profile
Screensaver
Trap
Unix Shell Configuration Modification
Windows Management Instrumentation Event Subscription
Abuse Elevation Control Mechanism (0/4)
Access Token Manipulation (0/5)
BITS Jobs
Build Image on Host
Debugger Evasion
Deobfuscate/Decode Files or Information
Deploy Container
Direct Volume Access
Domain Policy Modification (0/2)
Execution Guardrails (0/1)
Exploitation for Defense Evasion
File and Directory Permissions Modification (2/2)
Linux and Mac File and Directory Permissions Modification
Windows File and Directory Permissions Modification
Hide Artifacts (0/10)
Hijack Execution Flow (0/12)
Impair Defenses (0/10)
Indicator Removal (0/9)
Indirect Command Execution
Masquerading (0/8)
Modify Authentication Process (0/8)
Modify Cloud Compute Infrastructure (0/4)
Modify Registry
Modify System Image (0/2)
Network Boundary
Adversary-in-the-Middle (0/3)
Brute Force (0/4)
Credentials from Password Stores (0/5)
Exploitation for Credential Access
Forced Authentication
Forge Web Credentials (0/2)
Input Capture (0/4)
Modify Authentication Process (0/8)
Multi-Factor Authentication Interception
Multi-Factor Authentication Request Generation
Network Sniffing
OS Credential Dumping (0/8)
Steal Application Access Token
Steal or Forge Authentication Certificates
Steal or Forge Kerberos Tickets (0/4)
Steal Web Session Cookie
Unsecured Credentials (0/8)
Account Discovery (0/4)
Application Window Discovery
Browser Information Discovery
Cloud Infrastructure Discovery
Cloud Service Dashboard
Cloud Service Discovery
Cloud Storage Object Discovery
Container and Resource Discovery
Debugger Evasion
Device Driver Discovery
Domain Trust Discovery
File and Directory Discovery
Group Policy Discovery
Network Service Discovery
Network Share Discovery
Network Sniffing
Password Policy Discovery
Peripheral Device Discovery
Permission Groups Discovery (0/3)
Process Discovery
Query Registry
Remote System


Event Triggered Execution (7/16)
External Remote Services
Hijack Execution Flow (0/12)
Implant Internal Image
Modify Authentication Process (0/8)
Office Application Startup (0/6)
Pre-OS Boot (0/5)
Scheduled Task/Job (1/5)
Server Software Component (0/5)
Traffic Signaling (1/2)
Valid Accounts (0/4)
AppInit DLLs
Application Shimming
Change Default File Association
Component Object Model Hijacking
Emond
Image File Execution Options Injection
Installer Packages
LC_LOAD_DYLIB Addition
Netsh Helper DLL
PowerShell Profile
Screensaver
Trap
Unix Shell Configuration Modification
Windows Management Instrumentation Event Subscription
At
Container Orchestration Job
Cron
Scheduled Task
Systemd Timers
Port Knocking
Socket Filters
Exploitation for Privilege Escalation
Hijack Execution Flow (0/12)
Process Injection (4/12)
Scheduled Task/Job (1/5)
Valid Accounts (0/4)
Netsh Helper DLL
PowerShell Profile
Screensaver
Trap
Unix Shell Configuration Modification
Windows Management Instrumentation Event Subscription
At
Container Orchestration Job
Cron
Scheduled Task
Systemd Timers
Modify Authentication Process (0/8)
Modify Cloud Compute Infrastructure (0/4)
Modify Registry
Modify System Image (0/2)
Network Boundary Bridging (0/1)
Obfuscated Files or Information (0/11)
Plist File Modification
Pre-OS Boot (0/5)
Process Injection (0/12)
Reflective Code Loading
Rogue Domain Controller
Rootkit
Subvert Trust Controls (0/6)
System Binary Proxy Execution (0/13)
System Script Proxy Execution (0/1)
Template Injection
Traffic Signaling (1/2)
Trusted Developer Utilities Proxy Execution (0/1)
Unused/Unsupported Cloud Regions
Use Alternate Authentication Material (0/4)
Valid Accounts (0/4)
Virtualization/Sandbox Evasion (0/3)
Weaken Encryption (0/2)
XSL Script Processing
Authentication Certificates
Steal or Forge Kerberos Tickets (0/4)
Steal Web Session Cookie
Unsecured Credentials (0/8)
Peripheral Device Discovery
Permission Groups Discovery (0/3)
Process Discovery
Query Registry
Remote System Discovery
Software Discovery (0/1)
System Information Discovery
System Location Discovery (0/1)
System Network Configuration Discovery (0/1)
System Network Connections Discovery
System Owner/User Discovery
System Service Discovery
System Time Discovery
Virtualization/Sandbox Evasion (0/3)

`

Performed successfully
Failure to perform

[MITRE ATT&CK navigator map]