INTERNSHIP REPORT

-Tarun Kaushik, Computer Science Undergraduate

Attack and Breach Simulation Framework

Organization: BhumiiTech Pvt. Ltd.

Internship Duration: 1 month

Date: 15 May 2023- 15 June 2023

Table of Contents

Executive Summary

During my internship at BhumiiTech I have been actively involved in the development and implementation of an attack and simulation framework. My primary focus has been on studying and applying the MITRE ATT&CK framework and exploring the functionalities of the Atomic Red Team testing suite.

To begin, I conducted in-depth research on the MITRE ATT&CK framework, a comprehensive knowledge base of adversary tactics and techniques. I familiarized myself with its structure, sub-techniques, and the associated threat groups. This enabled me to understand the landscape of modern cyber threats and the various attack vectors used by adversaries. Building upon my knowledge of the MITRE ATT&CK framework, I delved into the Atomic Red Team project, an open-source testing suite designed to validate the effectiveness of security controls and identify potential weaknesses in an organization's defences. I thoroughly studied the documentation, including the available test cases and methodologies, to gain a solid understanding of how to simulate different attack techniques.

In order to gain practical experience, I successfully installed the Atomic Red Team framework on my Windows and kali Linux desktop machine. This involved setting up the necessary dependencies, configuring the environment, and ensuring compatibility with the target operating system. By doing so, I gained hands-on experience in utilizing the framework and executing simulated attacks within a controlled environment. Throughout this process, I documented my findings, observations, and any challenges encountered during the installation and usage of the Atomic Red Team framework. I also noted potential improvements and recommendations for optimizing its usability and functionality.

Research Description

1. Study the MITRE ATT&CK Framework so that the various TTPs can be well understood.

2. Install Atomic Red Team in systems Virtual Machine

3. Understand the functionality of the complete framework and demonstration of the same.

4. Throughout the project, effective documentation and reporting was maintained. To ensure clear communication and tracking of progress and also serve as valuable references for future analysis and improvement.

5. The report presented includes comprehensive information about the research conducted, scenario design, scenario development, proof of concept (POC), detailed network infrastructure, and mitigation strategies. And also provided the documentation of the findings, observations, adjustments made, and recommendations for each phase.

6. By adhering to a consistent reporting format and preparing detailed reports showcased the understanding and expertise and also created a valuable knowledge base for future reference and continuous improvement in the field of cybersecurity Red Teaming.

Methodology

1. Selection of Atomic Red Team Framework:

- Thoroughly researched and reviewed the Atomic Red Team framework, understanding its purpose, objectives, and capabilities.
- Assessed the compatibility of the framework with the organization's environment and security goals.

2. Planning and Objective Setting:

- Collaborated with the team and supervisor to define the scope and objectives of the internship project.
- Established specific goals for implementing Atomic Red Team using Kali Linux Purple and Windows 11.

3. Setup of Testing Environment:

- Configured a dedicated testing environment, isolated from production systems, to safely conduct the Atomic Red Team testing.
- Deployed Kali Linux Purple and Windows 11 virtual machines on separate hardware resources.

4. Identification of Security Controls:

- Collaborated with the organization's security team to identify the security controls to be tested.
- Conducted a thorough review of the existing security infrastructure, including antivirus, intrusion detection systems, firewall rules, and user access controls.

5. Selection and Customization of Atomic Tests:

- Explored the Atomic Red Team repository and identified a comprehensive set of Atomic Tests relevant to the identified security controls.

- Customized the Atomic Tests to align with the organization's specific environment, ensuring they targeted the intended systems and infrastructure.

6. Execution of Atomic Tests:

- Executed the adapted Atomic Tests on the testing environment, following proper guidelines and ethical considerations.
- Monitored and documented the results of each test, including successes, failures, and any unexpected behaviour.

7. Analysis and Evaluation:

- Analysed the outcomes of the Atomic Red Team testing, assessing the effectiveness of the security controls in detecting and mitigating the simulated attacks.
- Evaluated the performance of Kali Linux Purple and Windows 11 in executing the specific Atomic Tests, identifying any platform-specific vulnerabilities or limitations.

Implementation

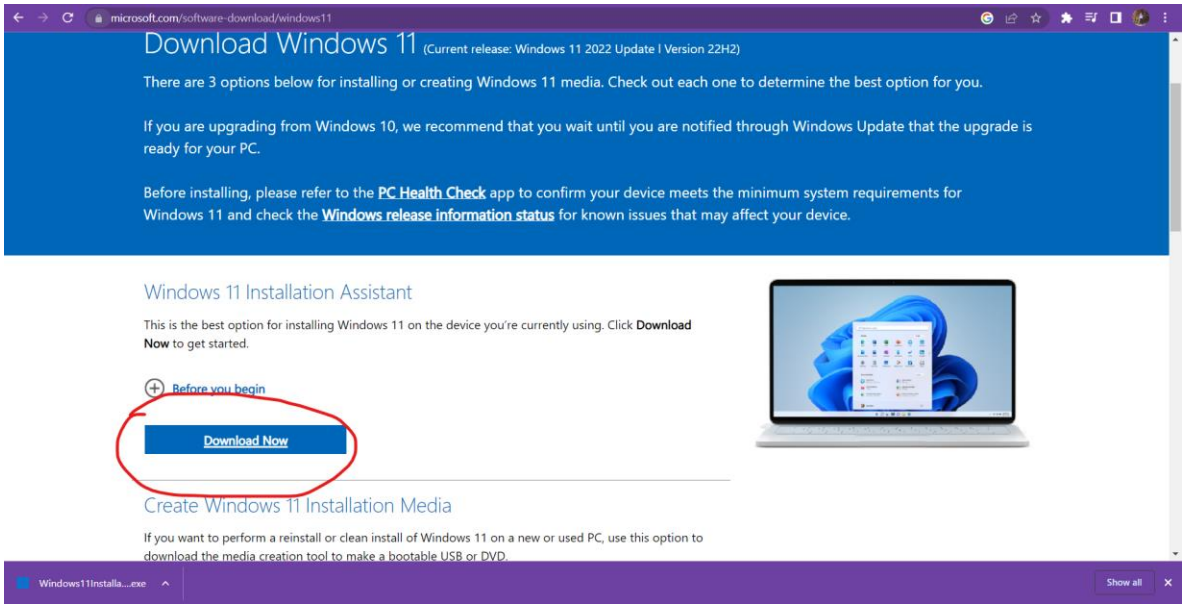
Infrastructure Setup:

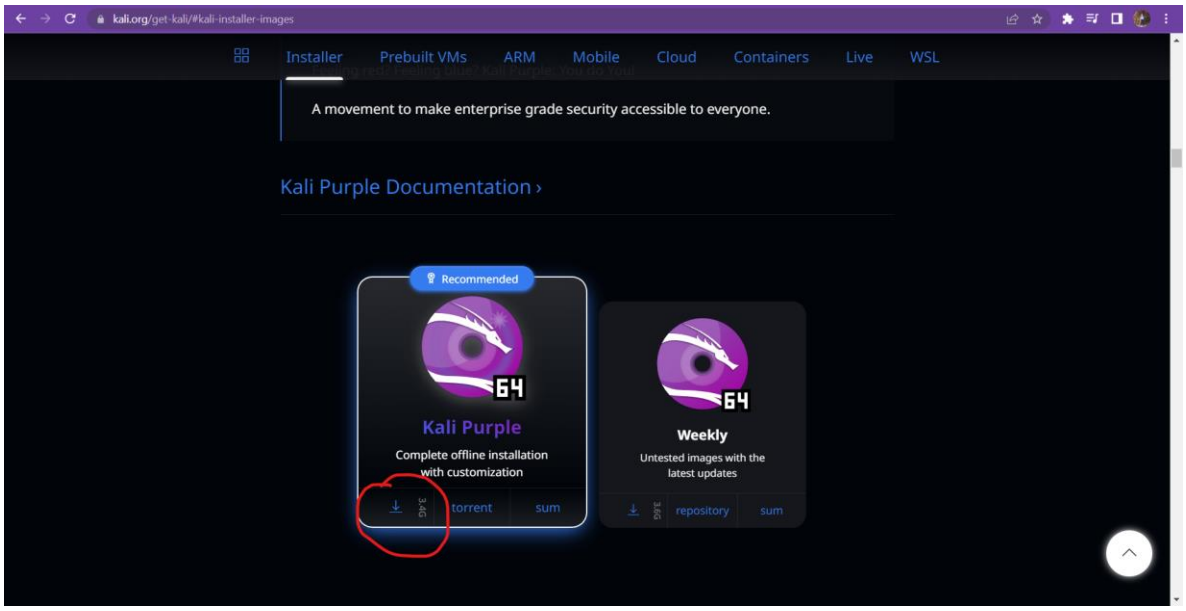
Installing Kali Linux Purple 2023.1 and Windows 11 VMs on VMware Workstation 17

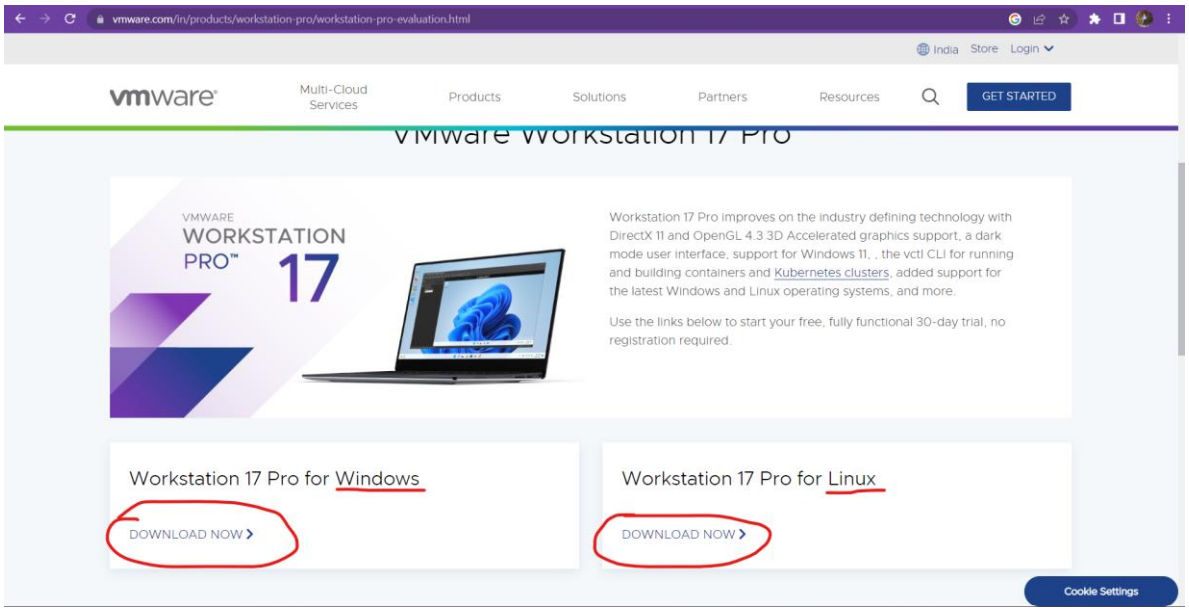
Steps involved:

1. Download VMware Workstation 17: Visit the official VMware website and download the latest version of VMware Workstation 17 compatible with your operating system.
2. Obtain Kali Linux Purple and Windows 11 ISOs: Download the ISO images for Kali Linux Purple2023.1 and Windows 11 from their respective official websites or trusted sources.
3. Create a New Virtual Machine (VM): Open VMware Workstation and click on "Create a New Virtual Machine." Choose the "Typical" configuration option.
4. Select Guest Operating System: In the "Guest Operating System Installation" window, select the appropriate option based on the operating system being installed. For Kali Linux Purple, select "Linux" and choose the correct version. For Windows 11, select "Windows" and choose the appropriate version.
5. Choose the ISO Image: In the next window, browse and select the Kali Linux Purple ISO file you downloaded earlier. Click "Next."
6. Configure Virtual Machine: Provide a name for the VM, choose the location where it will be stored, and set the disk size. Follow the prompts to complete the configuration settings, such as the number of processors, memory allocation, and network settings.

7. Customize Hardware: Review the hardware configuration and make any necessary adjustments. For example, increase the allocated RAM, enable virtualization extensions, or add additional virtual disks for storage. Ensure that the virtual network adapters are correctly configured.
8. Repeat Steps 3-7 for Windows 11: Create a new VM following the same steps mentioned above, but this time select the Windows 11 ISO file and configure the VM settings accordingly.
9. Start the VMs and Install the Operating Systems: Power on the Kali Linux Purple VM and follow the installation wizard to install the operating system. Repeat the process for the Windows 11 VM. Provide the required information during the installation process, such as the language, time zone, and account credentials.
10. Install VMware Tools: After the operating systems are installed, install VMware Tools on each VM to enhance the performance and functionality. In VMware Workstation, go to the "VM" menu, select "Install VMware Tools," and follow the on-screen instructions within each VM to complete the installation.
11. Customize and Configure: Customize the Kali Linux Purple and Windows 11 VMs based on your organization's requirements. This may include installing additional software, configuring network settings, and applying security updates.


microsoft.com/software-download/windows11
Download Windows 11 (Current release: Windows 11 2022 Update | Version 22H2)
There are 3 options below for installing or creating Windows 11 media. Check out each one to determine the best option for you.
If you are upgrading from Windows 10, we recommend that you wait until you are notified through Windows Update that the upgrade is ready for your PC.
Before installing, please refer to the PC Health Check app to confirm your device meets the minimum system requirements for Windows 11 and check the Windows release information status for known issues that may affect your device.
Windows 11 Installation Assistant
This is the best option for installing Windows 11 on the device you're currently using. Click Download Now to get started.
Before you begin
Download Now
Create Windows 11 Installation Media
If you want to perform a reinstall or clean install of Windows 11 on a new or used PC, use this option to download the media creation tool to make a bootable USB or DVD.
Windows11Installa....exe
Show all


kali.org/get-kali/#kali-installer-images
Installer
Prebuilt VMs
ARM
Mobile
Cloud
Containers
Live
WSL
A movement to make enterprise grade security accessible to everyone.
Kali Purple Documentation ›
Recommended
64
Kali Purple
Complete offline installation with customization
torrent
sum
64
Weekly
Untested images with the latest updates
repository
sum


vmware.com/in/products/workstation-pro/workstation-pro-evaluation.html
India
Store
Login
vmware
Multi-Cloud Services
Products
Solutions
Partners
Resources
GET STARTED
VMware Workstation 17 Pro
VMWARE
WORKSTATION PRO™ 17
Workstation 17 Pro improves on the industry defining technology with DirectX 11 and OpenGL 4.3 3D Accelerated graphics support, a dark mode user interface, support for Windows 11, , the vctl CLI for running and building containers and Kubernetes clusters, added support for the latest Windows and Linux operating systems, and more.
Use the links below to start your free, fully functional 30-day trial, no registration required.
Workstation 17 Pro for Windows
DOWNLOAD NOW ›
Workstation 17 Pro for Linux
DOWNLOAD NOW ›
Cookie Settings

Why use two different Operating systems?

Diverse Testing Environments:

Kali Linux and Windows 11 represent two distinct operating system environments commonly found in real-world scenarios. By utilizing both, you can simulate a wider range of target systems, applications, and network configurations, enabling more realistic and comprehensive testing.

Compatibility with Target Systems:

Different organizations and industries may have varying system architectures, with some predominantly using Linux-based systems (such as servers) while others rely on Windows-based systems (such as workstations). By having both Kali Linux and Windows 11 in your testing arsenal, you can better align with the target systems you are assessing, ensuring a more accurate evaluation of their security controls.

Diverse Attack Techniques:

Atomic Red Team covers a broad spectrum of attack techniques, including those specific to Linux and Windows environments. By having both operating systems available, you can execute and evaluate a wider array of predefined tests and custom attack scenarios, providing a more thorough assessment of the organization's defences.

Validation of Cross-Platform Defence:

Organizations often employ multi-layered security defences that span both Linux and Windows systems. By utilizing Kali Linux and Windows 11, you can validate the effectiveness of these cross-platform defence mechanisms, such as network-based intrusion detection systems, firewalls, or endpoint protection solutions, ensuring they can detect and mitigate attacks across different operating systems.

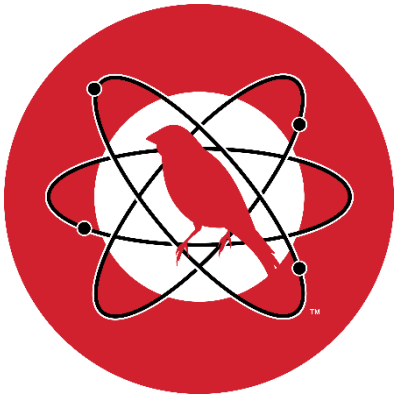
Flexibility and Adaptability:

Different security testing scenarios may require specific tools or techniques that are more readily available on one operating system than the other. By utilizing both Kali Linux and Windows 11, you can leverage the strengths and capabilities of each operating system, ensuring flexibility and adaptability to various testing requirements.

Enhanced Skill Development:

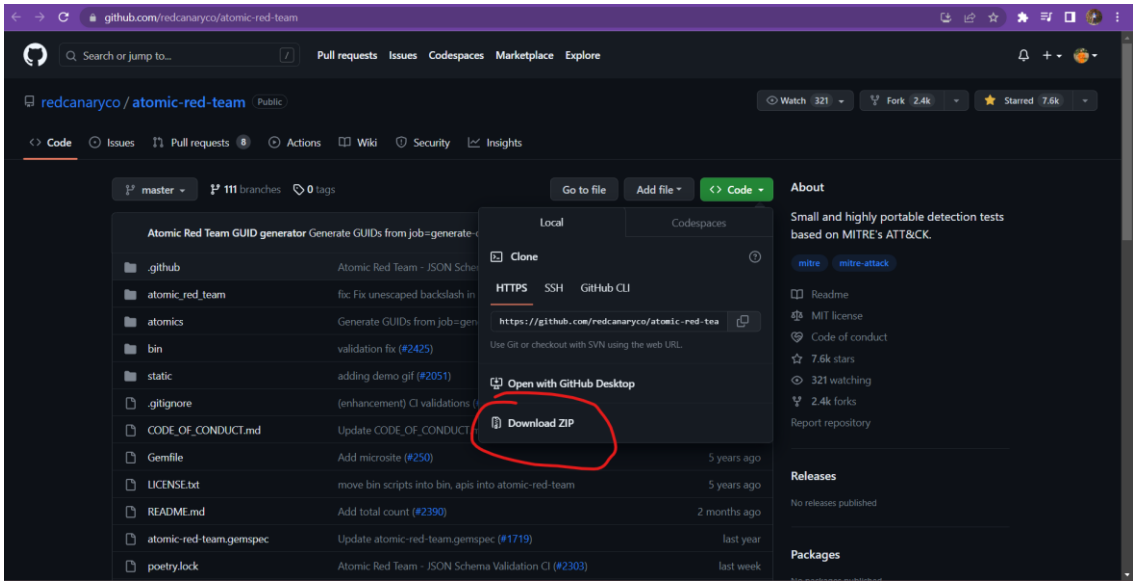
Working with both Kali Linux and Windows 11 provides an opportunity for security professionals to develop proficiency in different operating systems and associated security tools. This expanded skill set enables a better understanding of diverse security landscapes and enhances their ability to identify vulnerabilities and mitigate risks effectively.

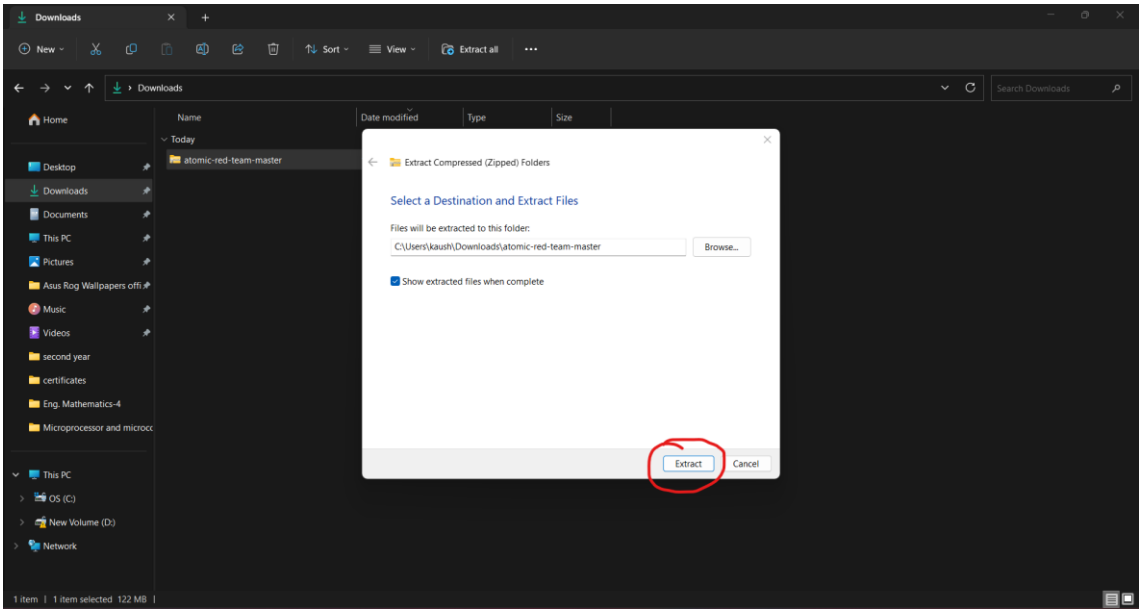
Installation of Atomic Red Team Framework:

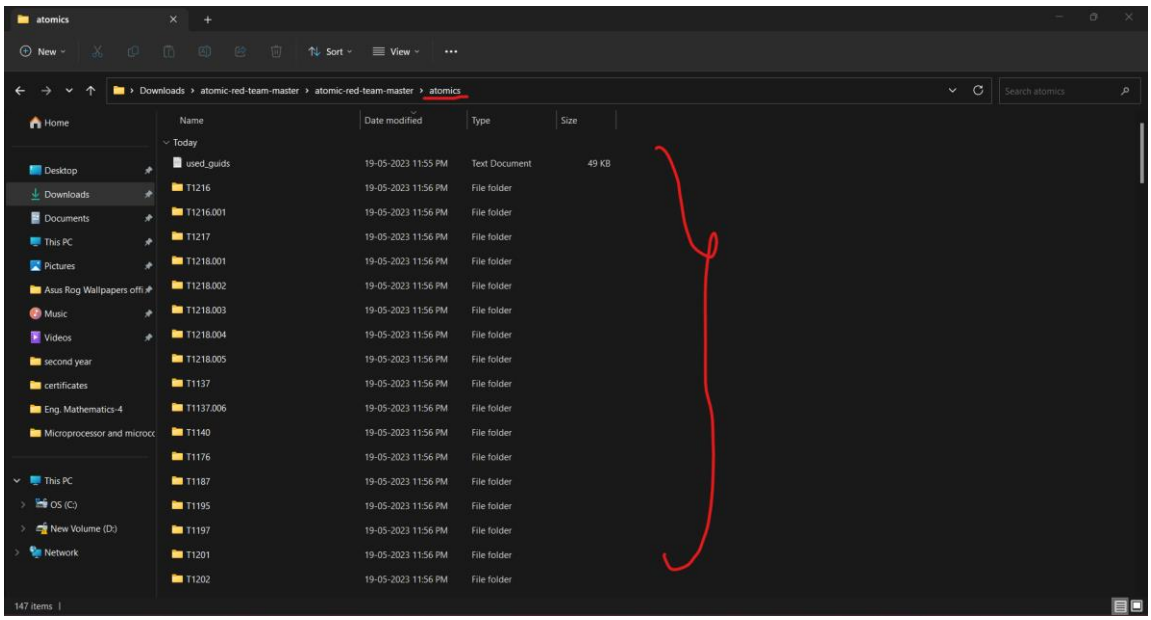


Steps Involved:

1. Prepare the Host System: Ensure that your host system meets the minimum requirements for running Atomic Red Team. This typically includes having a compatible operating system (such as Windows, Linux, or macOS) and sufficient resources like CPU, RAM, and disk space.
2. Install Dependencies: Before installing Atomic Red Team, ensure that all necessary dependencies are met. These dependencies may include programming languages (such as Python), libraries, frameworks, or specific tools required by Atomic Red Team. Refer to the official documentation or installation guide for the specific requirements.
3. Obtain Atomic Red Team: Download the Atomic Red Team framework from the official repository or trusted source. This can be obtained as a compressed archive or through a version control system like Git.
4. Extract the Archive: If you downloaded a compressed archive, extract its contents to a desired location on your host system. Use appropriate tools based on the archive format (e.g., tar, zip) to extract the files.
5. Configure Atomic Red Team: Navigate to the extracted directory and review the configuration options provided by Atomic Red Team. Modify the configuration files, if necessary, to customize the behaviour and settings according to your organization's requirements


github.com/redcanaryco/atomic-red-team
Search or jump to...
Pull requests
Issues
Codespaces
Marketplace
Explore
redcanaryco / atomic-red-team
Public
Watch 321
Fork 2.4k
Starred 7.6k
Code
Issues
Pull requests 8
Actions
Wiki
Security
Insights
master
111 branches
0 tags
Go to file
Add file
Code
Local
Codespaces
Clone
HTTPS
SSH
GitHub CLI
https://github.com/redcanaryco/atomic-red-tea
Use Git or checkout with SVN using the web URL.
Open with GitHub Desktop
Download ZIP
Atomic Red Team GUID generator Generate GUIDs from job=generate-
.github
atomic_red_team
atomics
bin
static
.gitignore
CODE_OF_CONDUCT.md
Gemfile
LICENSE.txt
README.md
atomic-red-team.gemspec
poetry.lock
Add microsite (#250)
5 years ago
move bin scripts into bin, apis into atomic-red-team
5 years ago
Add total count (#2390)
2 months ago
Update atomic-red-team.gemspec (#1719)
last year
Atomic Red Team - JSON Schema Validation CI (#2303)
last week
About
Small and highly portable detection tests based on MITRE's ATT&CK.
mitre
mitre-attack
Readme
MIT license
Code of conduct
7.6k stars
321 watching
2.4k forks
Report repository
Releases
No releases published
Packages


Downloads
New
Sort
View
Extract all
Downloads
Search Downloads
Home
Desktop
Downloads
Documents
This PC
Pictures
Asus Rog Wallpapers offi
Music
Videos
second year
certificates
Eng. Mathematics-4
Microprocessor and microcc
This PC
OS (C:)
New Volume (D:)
Network
Name
Date modified
Type
Size
Today
atomic-red-team-master
Extract Compressed (Zipped) Folders
Select a Destination and Extract Files
Files will be extracted to this folder:
C:\Users\kaush\Downloads\atomic-red-team-master
Browse...
Show extracted files when complete
Extract
Cancel
1 item | 1 item selected 122 MB


atomics
New
Sort
View
Downloads › atomic-red-team-master › atomic-red-team-master › atomics
Search atomics
Home
Desktop
Downloads
Documents
This PC
Pictures
Asus Rog Wallpapers offi
Music
Videos
second year
certificates
Eng. Mathematics-4
Microprocessor and microcc
This PC
OS (C:)
New Volume (D:)
Network
Name
Date modified
Type
Size
Today
used_guids
19-05-2023 11:55 PM
Text Document
49 KB
T1216
T1216.001
T1217
T1218.001
T1218.002
T1218.003
T1218.004
T1218.005
T1137
T1137.006
T1140
T1176
T1187
T1195
T1197
T1201
T1202
19-05-2023 11:56 PM
File folder
147 items

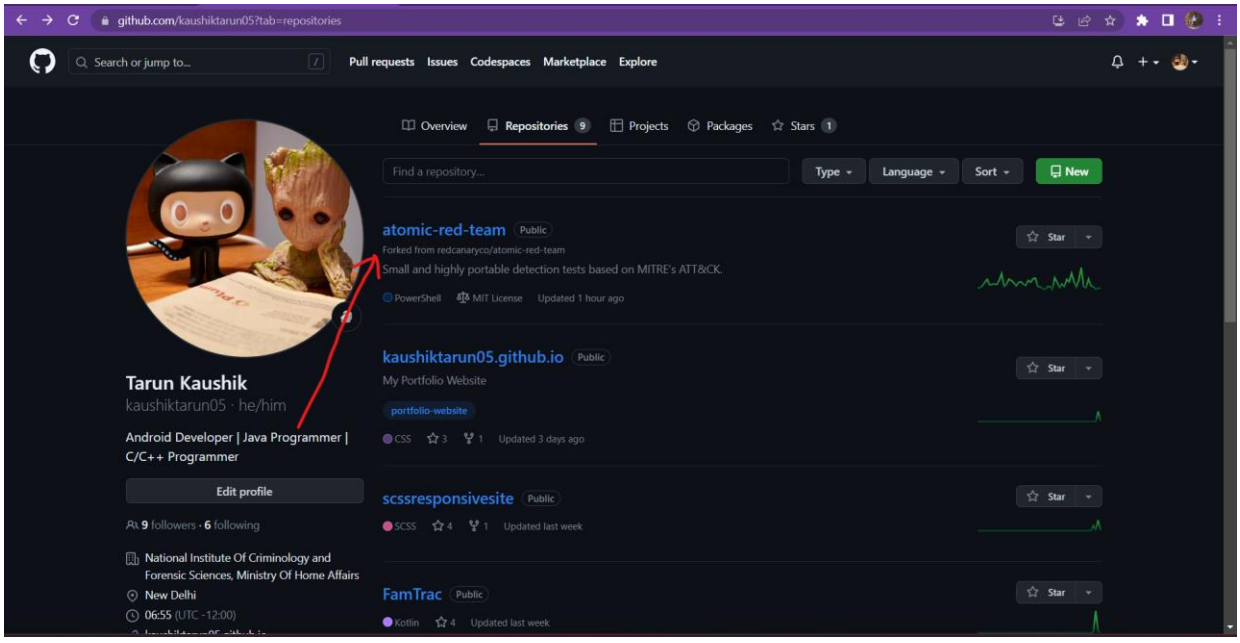
Clone and setup your repository in preparation for submitting a PR in GitHub

atomicredteam.txt

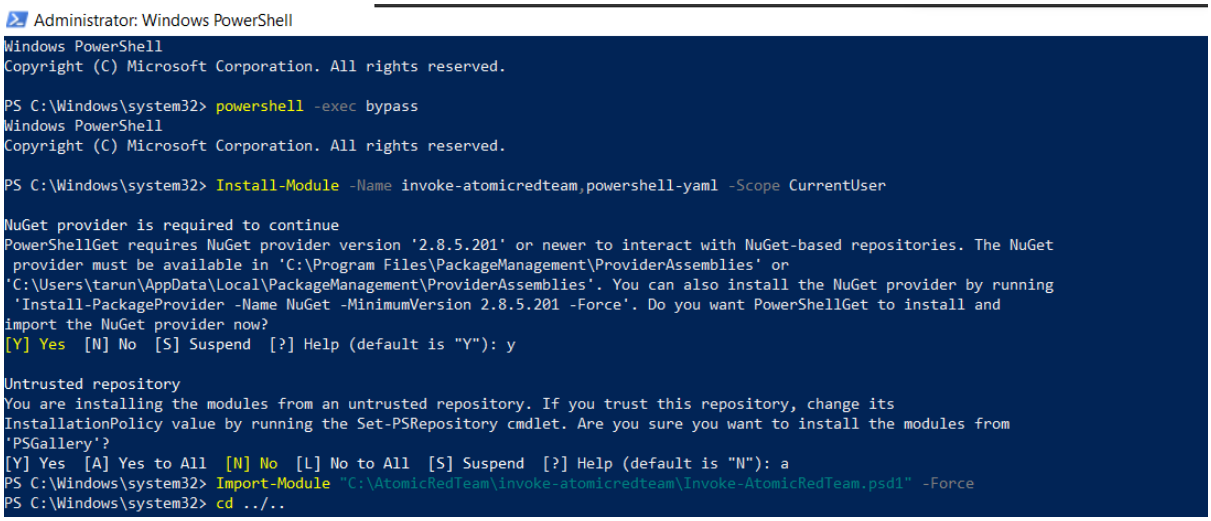
```
# Cloned fork of the Red Canary Atomic Red Team™ Repository
git clone https://github.com/mightyrock05/atomic-red-team.git

# Change directories into the cloned repository
cd atomic-red-team

# Set your origin (your fork) and your upstream (Red Canary's repo)
# You have to do this every time you re-clone your repo, which likely is not often
git remote set-url origin https://github.com/mightyrock05/atomic-red-team.git
git remote add upstream https://github.com/redcanaryco/atomic-red-team.git

# Update your forked master branch to match Red Canary's repo
# Do this right before creating a feature branch and working on it
git checkout master
git fetch --all
git rebase upstream/master
git push origin master

# Create a new branch from master to work on your new feature and switch to it
git checkout -b bhumiitech_test

# Add and commit your new/modified files to your local branch
git add /path/to/new/changed/file.yaml    # repeat for multiple files as needed
git commit -m "Changed has been commenced"

# Push the changes out to your repository residing in GitHub on the web
# The output from this command will tell you where to go on the web to submit the PR
git push origin bhumiitech_test
```

Note:

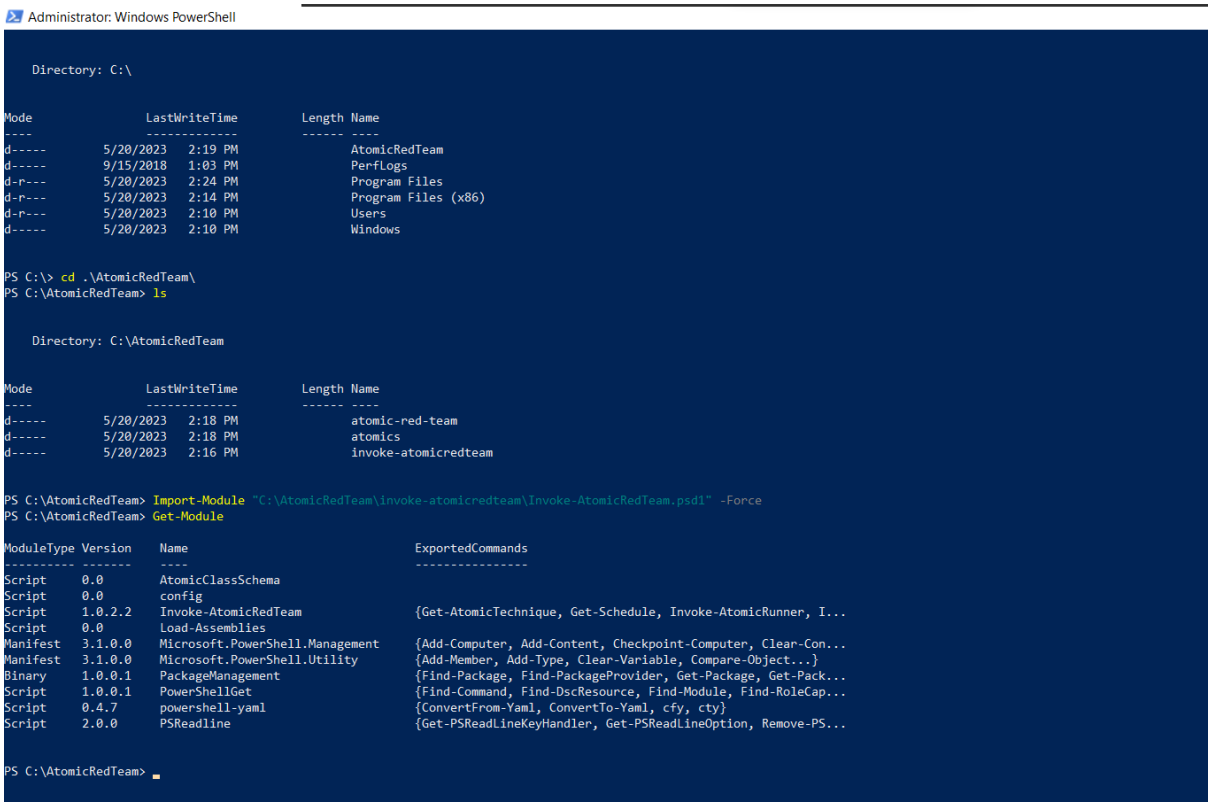
These git commands can be used on both our VM's i.e., on Kali Purple and Windows 11.



Setting up Atomic Red Team for attack



Importing and checking for Invoke Atomic red Module



RUNNING DIFFERENT TESTS ON SUSPECTED VULNERABLE MACHINES

Test T1016

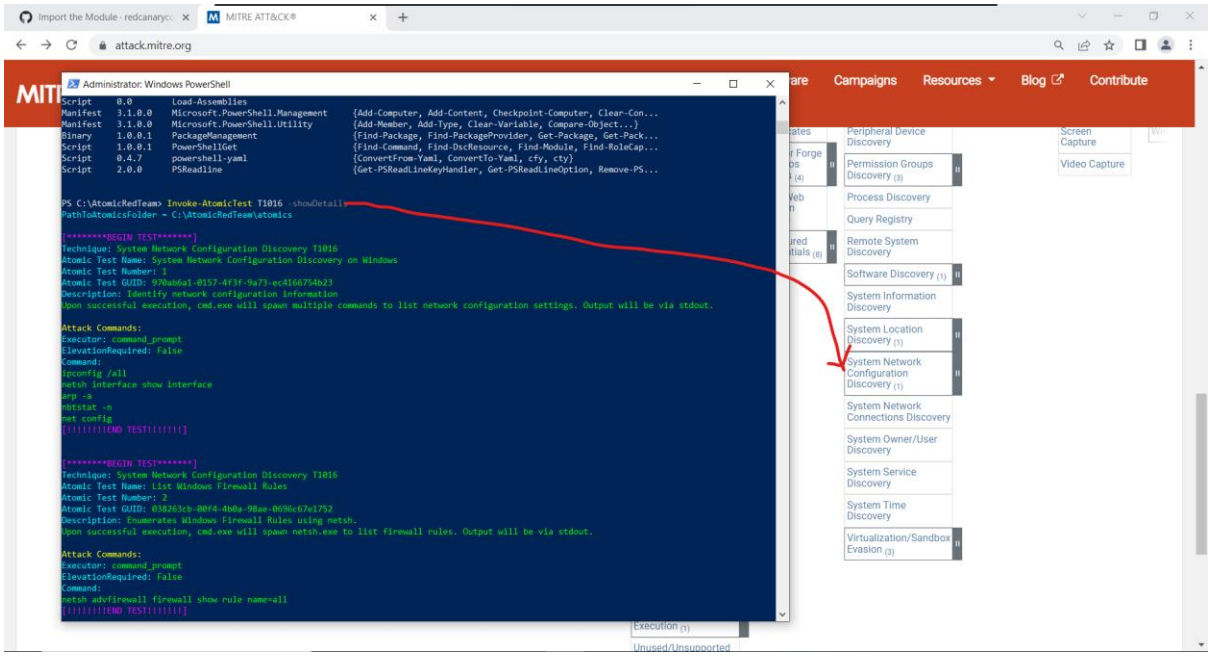
The purpose to run this test is to simulate an attacker's attempt to gather information about the network configuration of a target system. This test helps organizations assess their ability to detect and respond to such reconnaissance activities, enhancing their overall security posture.

The execution of test T1016 typically involves the following steps:

1. Initial Reconnaissance: The attacker initiates the reconnaissance phase by identifying a target system or network. This may involve various techniques such as open-source intelligence (OSINT) gathering, scanning, or identifying potential targets through phishing emails.
2. Network Scanning: The attacker performs network scanning to identify active hosts and open ports within the target network. This may include using tools like Nmap or custom scripts to probe the target system for available network services.
3. Service Enumeration: Once active hosts and open ports are identified, the attacker proceeds to enumerate the network services running on those systems. This may involve sending specific network requests to various ports to elicit responses from running services and determine their versions.
4. System Fingerprinting: With the knowledge of active network services, the attacker attempts to fingerprint the target system. This step involves gathering information about the operating system, software versions, and other system-specific details that can assist in further exploitation.

5. Network Mapping: Using the obtained information, the attacker constructs a network map or diagram, outlining the interconnected devices and their relationships within the target network. This mapping provides valuable insights into the network's architecture, potential vulnerabilities, and potential points of entry.
6. Data Collection: The attacker collects the gathered information, including the network map, system configurations, and other relevant details. This data serves as reconnaissance for further exploitation attempts or as a means to identify potential weaknesses that can be reported to the organization for remediation.

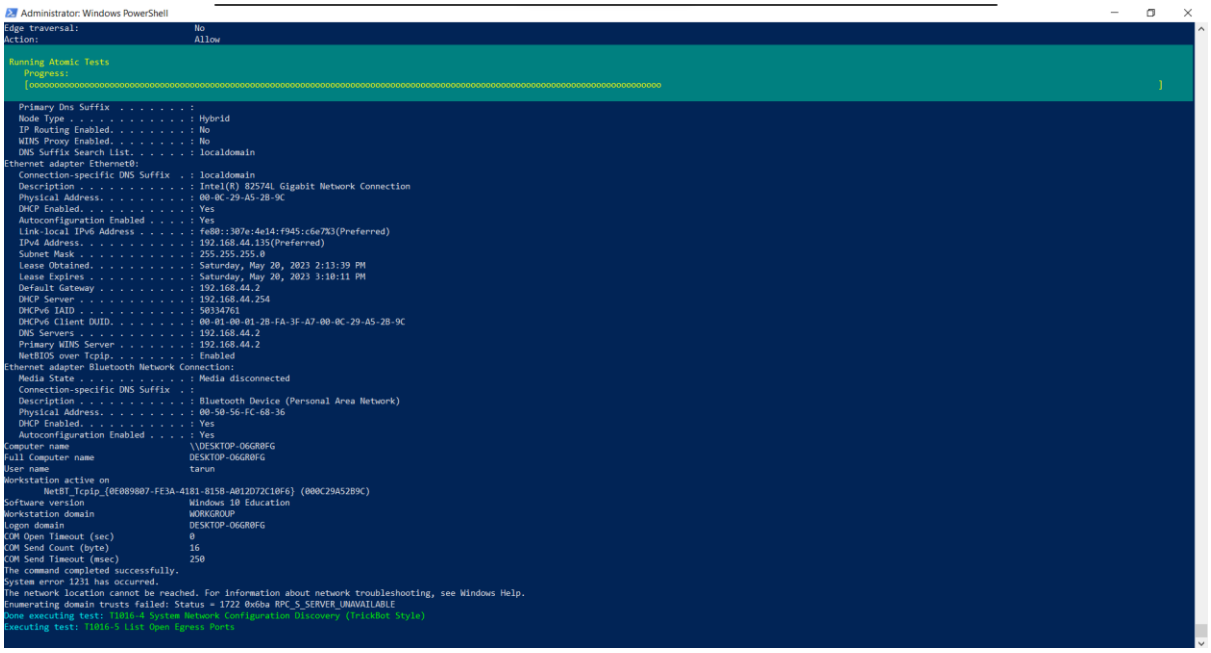
Test Details



Successful detection and response to this test demonstrate an organization's ability to identify reconnaissance activities and take appropriate actions to mitigate potential threats. It also helps organizations identify gaps in their

security controls and refine their incident response procedures to enhance their overall security posture.

Running Test T1016



By simulating these steps, security teams can assess the effectiveness of their monitoring and alerting systems, intrusion detection systems (IDS), intrusion prevention systems (IPS), and other security controls.

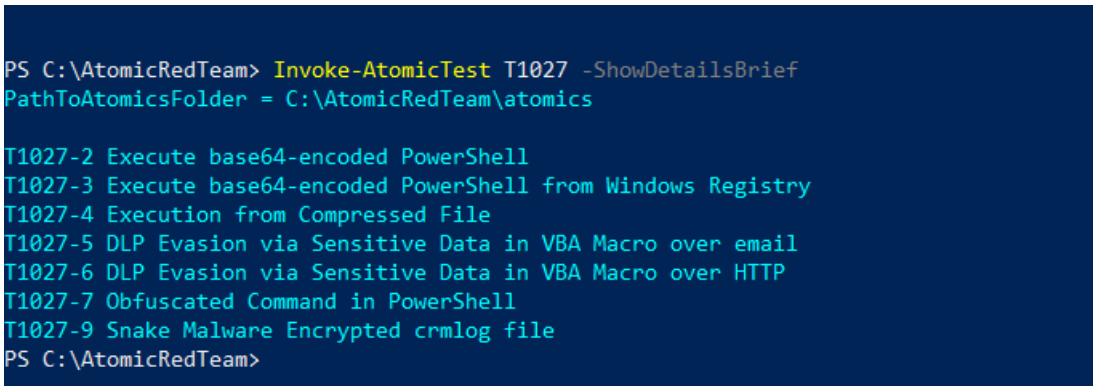
Test T1016: System Network Configuration Discovery - Test Results

During the execution of test T1016, all outcomes were observed as negative. This means that the organization's defensive capabilities did not detect or respond to the simulated network configuration discovery activities.

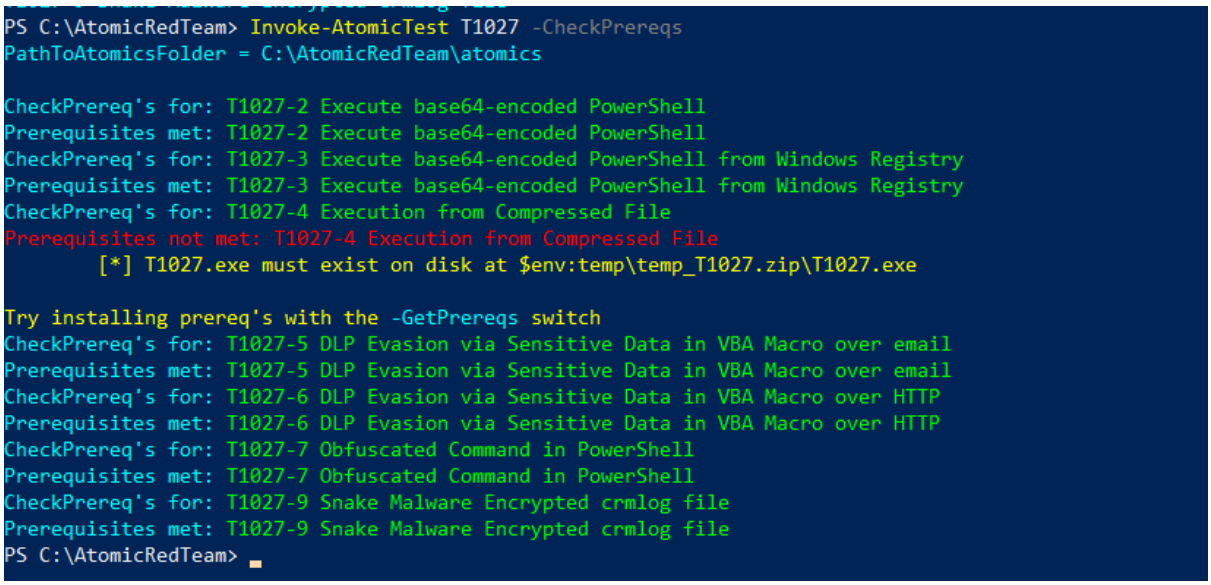
Test T1027

Test T1027 in Atomic Red Team is titled "Obfuscated Files or Information." Its purpose is to simulate an attacker's attempt to hide malicious files or information within a system using obfuscation techniques. This test helps organizations assess their ability to detect and respond to such obfuscated content, enhancing their overall security posture.

Test Details in Brief



Checking for prerequisites



Fulfilling prerequisites for test

```
PS C:\AtomicRedTeam> Invoke-AtomicTest T1027 -GetPrereqs
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

GetPrereq's for: T1027-2 Execute base64-encoded PowerShell
No Preqs Defined
GetPrereq's for: T1027-3 Execute base64-encoded PowerShell from Windows Registry
No Preqs Defined
GetPrereq's for: T1027-4 Execution from Compressed File
Attempting to satisfy prereq: T1027.exe must exist on disk at $env:temp\temp_T1027.zip\T1027.exe
Prereq successfully met: T1027.exe must exist on disk at $env:temp\temp_T1027.zip\T1027.exe
GetPrereq's for: T1027-5 DLP Evasion via Sensitive Data in VBA Macro over email
No Preqs Defined
GetPrereq's for: T1027-6 DLP Evasion via Sensitive Data in VBA Macro over HTTP
No Preqs Defined
GetPrereq's for: T1027-7 Obfuscated Command in PowerShell
No Preqs Defined
GetPrereq's for: T1027-9 Snake Malware Encrypted crmlog file
No Preqs Defined
PS C:\AtomicRedTeam>
```

The execution of test T1027 typically involves the following steps:

1. Obfuscation Technique Selection: The attacker selects an appropriate obfuscation technique to hide malicious files or information. This could include techniques such as encoding, encryption, packing, or using steganography to conceal data within seemingly innocuous files.
2. Payload Creation: The attacker crafts a payload containing the malicious files or information. The payload is obfuscated using the selected technique to make it difficult for traditional security tools to detect or analyse the content.
3. Delivery or Execution: The attacker delivers the obfuscated payload to the target system or executes it within the environment. This could occur through various methods, including email attachments, malicious downloads, or exploiting vulnerabilities in the system.
4. Decoding or De obfuscation: Upon delivery or execution, the obfuscated payload needs to be decoded or de obfuscated to reveal its true nature. The attacker leverages the obfuscation technique used to reverse the process and retrieve the original malicious files or information.

5. Malicious Activity: Once the obfuscated payload is decoded, the attacker's intended malicious activity takes place. This could include actions such as executing malicious code, establishing a backdoor, exfiltrating sensitive data, or further compromising the system's security.

Running the Test

```
PS C:\AtomicRedTeam> Invoke-AtomicTest T1027
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

Executing test: T1027-2 Execute base64-encoded PowerShell
VwByAGkAdAB1AC0ASABvAHMAdAAgACIASABlAHkALAAgAEEAdABvAG0AaQBjACEAIgA=
Hey, Atomic!
Hey, Atomic!
Done executing test: T1027-2 Execute base64-encoded PowerShell
Executing test: T1027-3 Execute base64-encoded PowerShell from Windows Registry
VwByAGkAdAB1AC0ASABvAHMAdAAgACIASABlAHkALAAgAEEAdABvAG0AaQBjACEAIgA=
Hey, Atomic!
Done executing test: T1027-3 Execute base64-encoded PowerShell from Windows Registry
Executing test: T1027-4 Execution from Compressed File
Done executing test: T1027-4 Execution from Compressed File
Executing test: T1027-5 DLP Evasion via Sensitive Data in VBA Macro over email
Send-MailMessage : Unable to connect to the remote server
At line:1 char:4
+ & {Send-MailMessage -From test@corp.com -To test@corp.com -Subject 'T ...
+    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidOperation: (System.Net.Mail.SmtpClient:SmtpClient) [Send-MailMessage], SmtpExcept
   ion
    + FullyQualifiedErrorId : SmtpException,Microsoft.PowerShell.Commands.SendMailMessage
Done executing test: T1027-5 DLP Evasion via Sensitive Data in VBA Macro over email
Executing test: T1027-6 DLP Evasion via Sensitive Data in VBA Macro over HTTP
Invoke-WebRequest : Unable to connect to the remote server
At line:1 char:4
+    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + FullyQualifiedErrorId : System.Net.WebException,Microsoft.PowerShell.Commands.InvokeWebRequestCommand
    + CategoryInfo          : NotSpecified: (:) [Invoke-WebRequest], WebException
+ & {Invoke-WebRequest -Uri 127.0.0.1 -Method POST -Body C:\AtomicRedTe ...
Done executing test: T1027-6 DLP Evasion via Sensitive Data in VBA Macro over HTTP
Executing test: T1027-7 Obfuscated Command in PowerShell
Hello, from PowerShell!
Done executing test: T1027-7 Obfuscated Command in PowerShell
Executing test: T1027-9 Snake Malware Encrypted crmlog file
File created: C:\Windows\registration\04e53197-72be-4dd8-88b1-533fe6eed577.04e53197-72be-4dd8-88b1-533fe6eed577.crmlog
Done executing test: T1027-9 Snake Malware Encrypted crmlog file
PS C:\AtomicRedTeam>
```

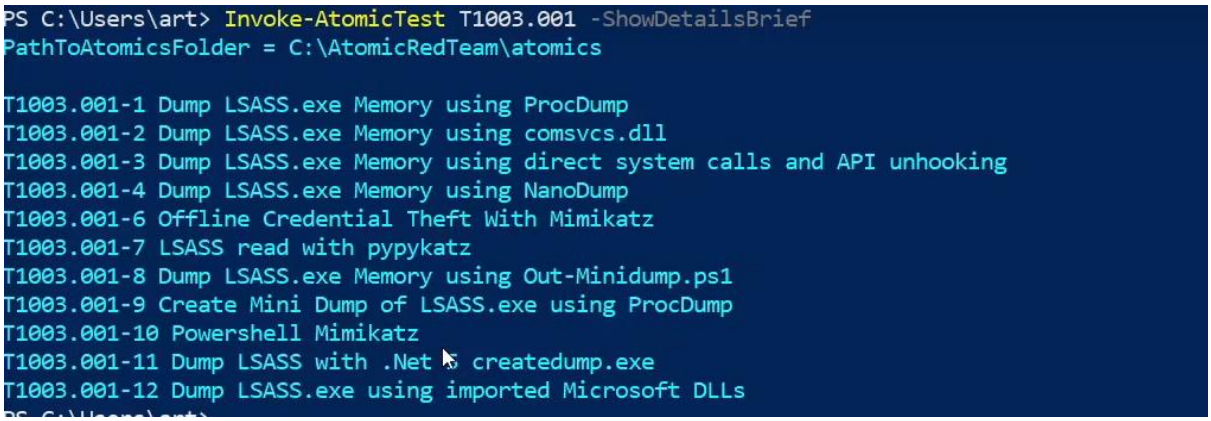
Test T1027: Obfuscated Files or Information - Test Results

During the execution of test T1027, all outcomes were observed as negative. This means that the organization's defensive capabilities did not detect or respond to the simulated obfuscated files or information.

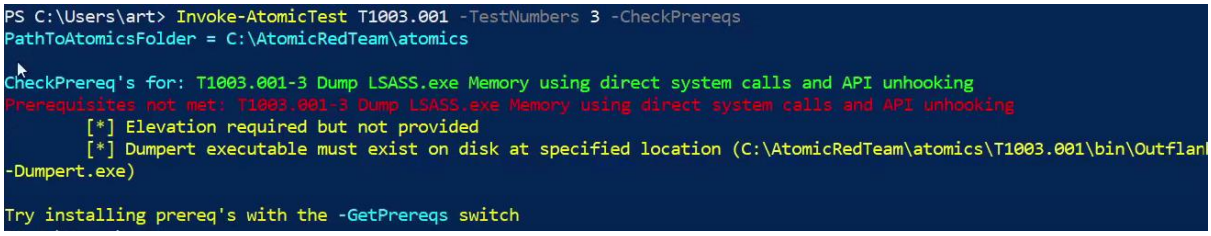
Test T1003.001

Test T1003 in Atomic Red Team is titled "OS Credential Dumping." Its purpose is to simulate an attacker's attempt to extract account credentials from an operating system. This test helps organizations assess their ability to detect and respond to credential dumping techniques, enhancing their overall security posture.

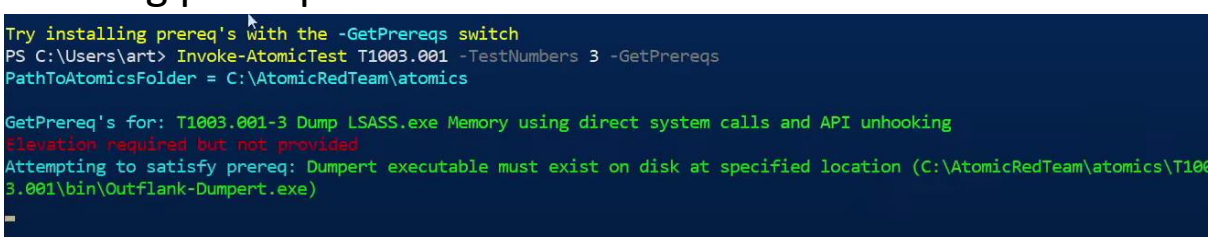
Test Details in Brief



Checking for prerequisites



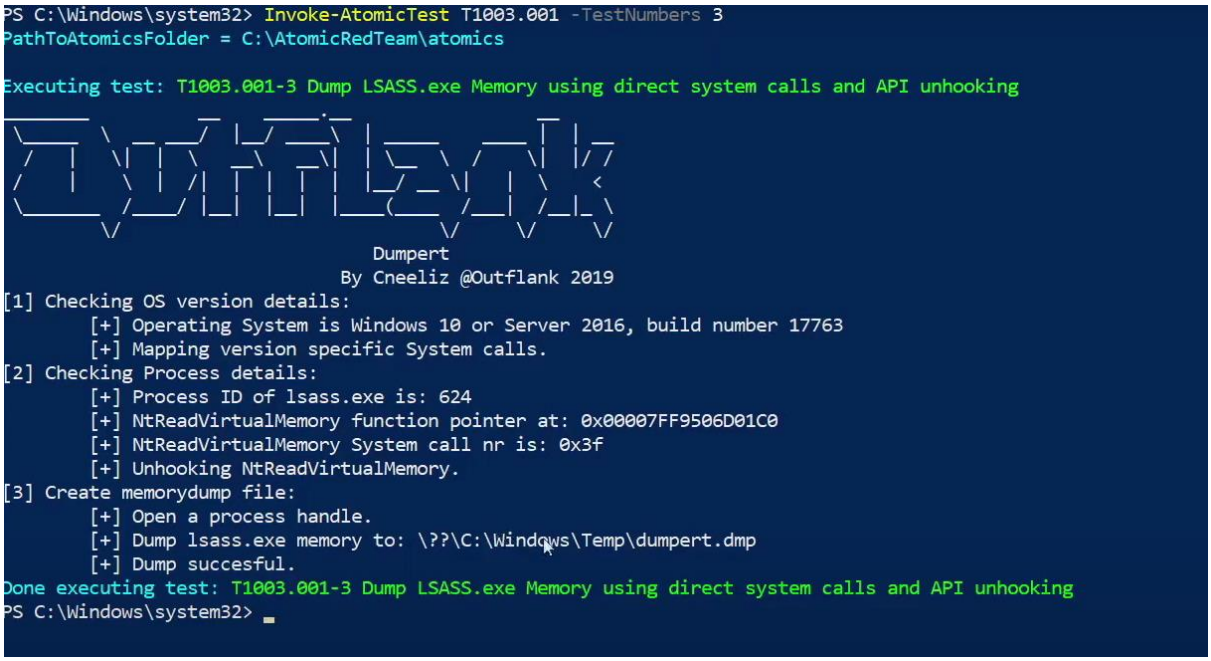
Fulfilling prerequisites for test



The execution of test T1003 typically involves the following steps:

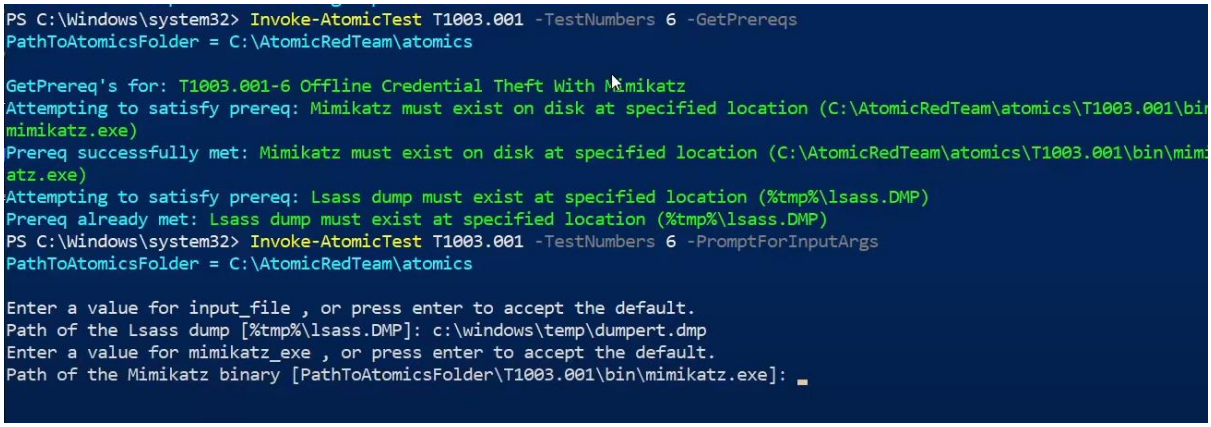
1. Initial Access: The attacker gains initial access to the target system through various methods, such as exploiting vulnerabilities, using stolen credentials, or leveraging social engineering techniques.
2. Privilege Escalation: Once inside the target system, the attacker attempts to escalate privileges to gain higher-level access, allowing them to access sensitive areas of the operating system.
3. Credential Dumping Technique Selection: The attacker selects an appropriate credential dumping technique based on the operating system and security controls in place. This could include techniques such as using tools like Mimi Katz, leveraging Windows Credential Editor (WCE), or extracting credentials from LSASS memory.
4. Execution of Credential Dumping Technique: The attacker executes the chosen credential dumping technique to extract account credentials from the operating system. This may involve running specific commands or using pre-compiled tools to extract passwords, hashes, or other authentication tokens.
5. Harvesting of Credentials: After successfully executing the credential dumping technique, the attacker collects the extracted credentials, which can include usernames, passwords, or other forms of authentication information.
6. Use of Harvested Credentials: With the harvested credentials in hand, the attacker can utilize them to gain unauthorized access to other systems, escalate privileges further, or perform other malicious activities within the network.

Running the test



Disabling the Windows defender to access the dumpert.dmp file

Getting access to all the password hash



Note:

We have to use the Test Number 6 in same T1003.001 to show hash credentials.

Test T1003: OS Credential Dumping - Test Results

During the execution of test T1003, the outcome was observed as positive, indicating that the attacker was able to obtain credential hashes of the user. This result highlights potential vulnerabilities in the organization's security infrastructure related to credential protection and detection.

Obtaining credential hashes can significantly impact an organization's security posture, as these hashes can be cracked and used to gain unauthorized access to systems or escalate privileges within the network. It is crucial to address this issue promptly to mitigate the risks associated with credential-based attacks.

To address this, it is recommended that the organization take the following steps:

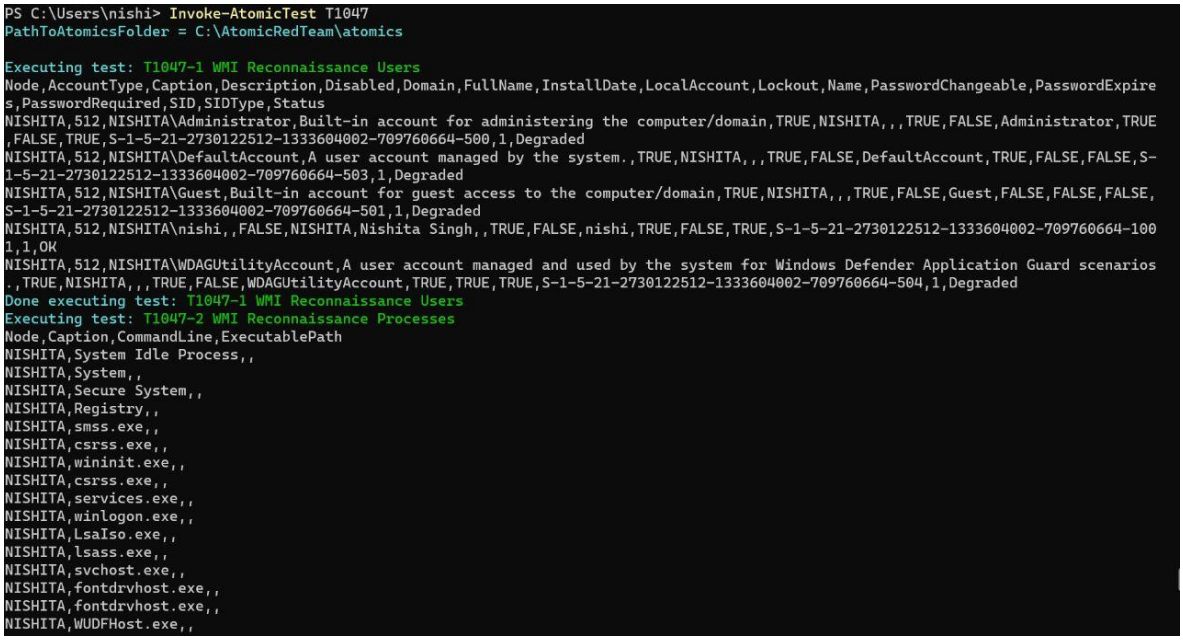
1. Implement Strong Authentication Measures: Enforce the use of strong, unique passwords and consider implementing multi-factor authentication (MFA) for all user accounts. MFA adds an extra layer of security by requiring additional verification beyond passwords, making it more challenging for attackers to exploit credential hashes.
2. Enhance Privileged Account Management: Implement privileged access management (PAM) solutions to manage and monitor privileged accounts effectively. This includes implementing password rotation policies, limiting the use of privileged accounts, and monitoring for any suspicious activity related to privileged access.
3. Patch and Update Systems: Regularly apply security patches and updates to operating systems and software to address known vulnerabilities that could be exploited for credential dumping. This helps mitigate the risk of privilege escalation and unauthorized access to sensitive information.
4. Implement Endpoint Detection and Response (EDR): Deploy EDR solutions that can detect and respond to credential dumping activities. These solutions leverage behavioural analysis and anomaly detection techniques to identify malicious activities and promptly initiate incident response procedures.

5. Monitor for Unusual Activity: Implement comprehensive log monitoring and analysis solutions to detect anomalous behaviour related to credential dumping. Monitor system logs, network traffic, and authentication events to identify potential indicators of compromise and respond accordingly.

Test T1047

Test T1047 in Atomic Red Team is titled "Windows Management Instrumentation (WMI) Persistence." Its purpose is to simulate an attacker's attempt to establish persistence on a Windows system using WMI. This test helps organizations assess their ability to detect and respond to malicious activities involving WMI persistence.

Test Details in Brief



The execution of test T1027 typically involves the following steps:

1. Initial Access: The attacker gains initial access to the target system through various methods, such as exploiting vulnerabilities, using stolen credentials, or leveraging social engineering techniques.
2. Privilege Escalation: Once inside the target system, the attacker attempts to escalate privileges to gain higher-level access, allowing them to modify WMI settings.

3. WMI Persistence Technique Selection: The attacker selects a specific technique to establish persistence using WMI. This may include creating a WMI event subscription, modifying WMI permanent event subscriptions, or using the WMI command-line tool (wmic) to create or modify WMI objects.
4. Execution of WMI Persistence Technique: The attacker executes the chosen WMI persistence technique to create a mechanism that will survive system reboots and remain active for future interactions. This involves leveraging WMI's capabilities to execute arbitrary commands, schedule tasks, or invoke malicious scripts.
5. Verifying Persistence: After establishing the WMI persistence mechanism, the attacker verifies that it is functioning as intended. This may involve checking if the event subscription or modified WMI object is active and operational.
6. Performing Malicious Actions: With the WMI persistence mechanism in place, the attacker can leverage it to perform malicious actions, such as executing commands, launching scripts, or downloading and executing additional payloads.

Running the Test

```
Done executing test: T1047-7 Create a Process using WMI Query and an Encoded Command
Executing test: T1047-8 Create a Process using obfuscated Win32_Process
Exception calling "Put" with "0" argument(s): "Access denied "
At line:3 char:1
+ $NewClass.Put()
+ ~~~~~~~~~~~~~~~
    + CategoryInfo          : NotSpecified: (:) [], MethodInvocationException
    + FullyQualifiedErrorId : DotNetMethodException

Invoke-WmiMethod : Not found
At line:4 char:1
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
+ Invoke-WmiMethod -Path Win32_Atomic -Name create -ArgumentList notepa ...
    + CategoryInfo          : InvalidOperation: (:) [Invoke-WmiMethod], ManagementException
    + FullyQualifiedErrorId : InvokeWMIManagementException,Microsoft.PowerShell.Commands.InvokeWmiMethod
Done executing test: T1047-8 Create a Process using obfuscated Win32_Process
Executing test: T1047-9 WMI Execute rundll32
Program 'WMIC.exe' failed to run: Operation did not complete successfully because the file contains a virus or
potentially unwanted softwareAt line:1 char:4
+ & {wmic /node:127.0.0.1 process call create "rundll32.exe $env:TEMP\c ...
At line:1 char:4
+    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~.
+ & {wmic /node:127.0.0.1 process call create "rundll32.exe $env:TEMP\c ...
+    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + FullyQualifiedErrorId : NativeCommandFailed
    + CategoryInfo          : ResourceUnavailable: (:) [], ApplicationFailedException
Done executing test: T1047-9 WMI Execute rundll32
Executing test: T1047-10 Application uninstall using WMIC
No Instance(s) Available.
Done executing test: T1047-10 Application uninstall using WMIC
```

Test T1047: Windows Management Instrumentation (WMI) Persistence - Test Results

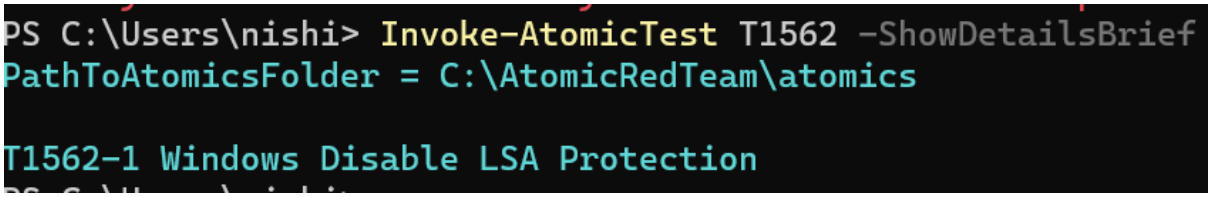
During the execution of test T1047, all outcomes were observed as negative, indicating that the organization has demonstrated the capability to detect and respond to malicious activities involving WMI persistence. This result is an encouraging sign of the organization's effective security controls and monitoring mechanisms.

Detecting and responding to WMI persistence is crucial for mitigating the risks associated with attackers establishing long-term access and control over compromised systems. By achieving negative outcomes in this test, the organization has shown that it has implemented appropriate measures to monitor and detect suspicious activities related to WMI persistence.

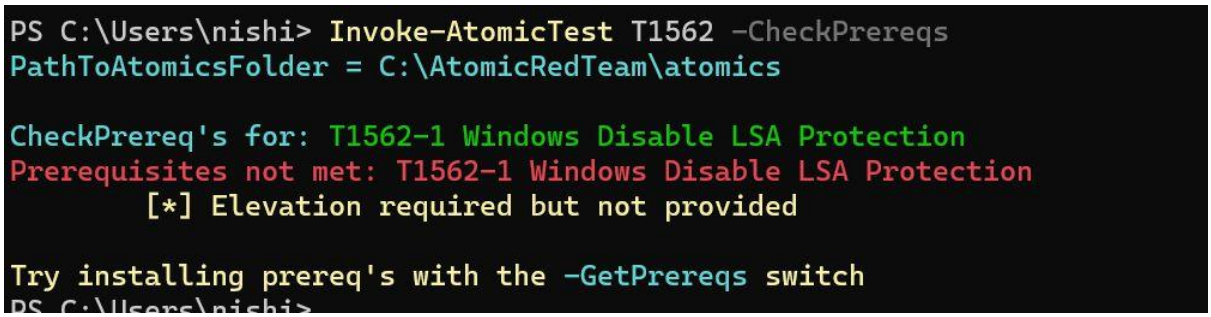
Test T1562

Test T1562 in Atomic Red Team is titled "Impersonation: Pass the Hash." The purpose of this test is to simulate an attacker leveraging stolen password hashes to authenticate and gain unauthorized access to a system or network. The execution of this test involves using the pass-the-hash technique, where the attacker uses the hash of a user's password instead of the actual plaintext password.

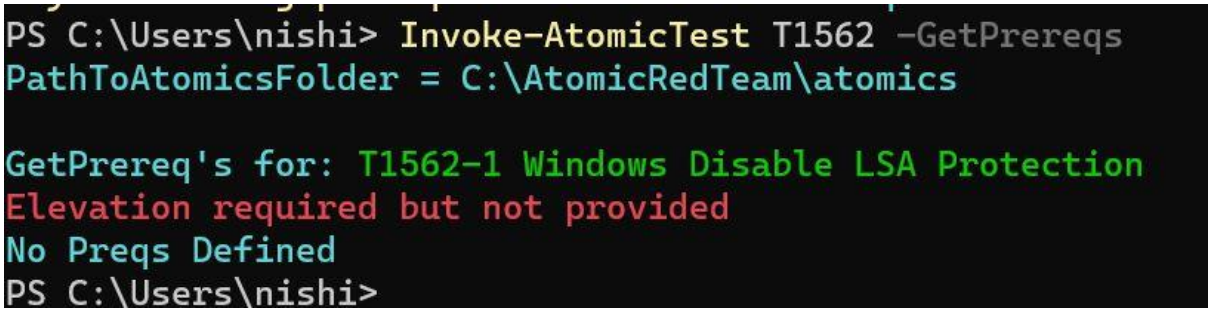
Test Details in Brief



Checking for prerequisites



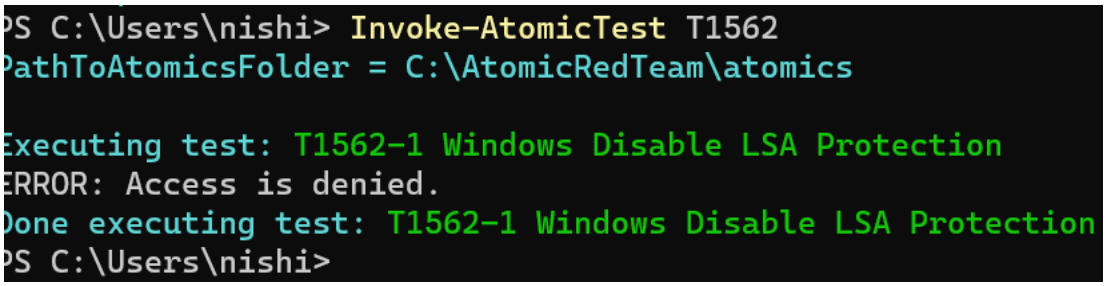
Fulfilling prerequisites for test



The complete execution of test T1562 typically involves the following steps:

1. Initial Access: The attacker gains initial access to the target system or network through various methods, such as exploiting vulnerabilities, using stolen credentials, or employing social engineering techniques.
2. Password Hash Extraction: The attacker extracts password hashes from compromised systems or domains. This can be achieved through techniques like password dumping tools, credential harvesting, or exploiting vulnerabilities that expose password hashes.
3. Pass-the-Hash Attack: The attacker uses the extracted password hashes to impersonate a legitimate user and gain unauthorized access. Instead of cracking the hashes to obtain the actual passwords, the attacker directly uses the hash value during the authentication process.
4. Authentication and Access: The attacker leverages the stolen password hashes to authenticate to the target system or network. If successful, they gain access with the privileges associated with the compromised user's account.
5. Lateral Movement and Persistence: Once authenticated, the attacker may perform lateral movement by accessing additional systems within the network. They can exploit the compromised user's privileges to escalate access, gain control over critical systems, or establish persistence within the network.
6. Detection and Response Assessment: Monitor the target environment and assess the effectiveness of detection and response mechanisms. This includes observing if the pass-the-hash attack is detected through anomaly detection, behaviour-based monitoring, or security solutions specifically designed to detect credential misuse.

RUNNING THE TEST



Test T1562: Impersonation - Pass the Hash - Test Results

During the execution of test T1562, all outcomes were observed as "Error: Access Denied." This indicates that the organization has implemented effective security measures to prevent pass-the-hash attacks and unauthorized access using stolen password hashes.

The "Access Denied" result suggests that the organization has implemented strong authentication controls, privileged access management, and monitoring mechanisms that are capable of detecting and preventing the misuse of stolen password hashes. This is a positive outcome, highlighting the organization's commitment to securing sensitive credentials and mitigating the risks associated with pass-the-hash attacks.

Test T1059

Test T1059 in Atomic Red Team is titled "Command-Line Interface (CLI)." The purpose of this test is to simulate an attacker executing commands and performing malicious actions through a command-line interface on a target system. The execution of this test involves leveraging various CLI tools and techniques to assess the effectiveness of security controls and monitoring mechanisms.

Test Details in Brief

```
PS C:\Users\nishi> Invoke-AtomicTest T1059 -ShowDetailsBrief
PathToAtomicsFolder = C:\AtomicRedTeam\atomics
```

Checking for prerequisites

```
PS C:\Users\nishi> Invoke-AtomicTest T1059 -CheckPrereqs
PathToAtomicsFolder = C:\AtomicRedTeam\atomics
```

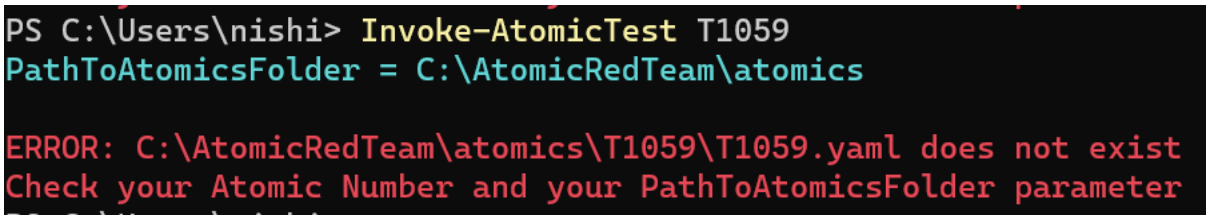
Fulfilling prerequisites for test

```
PS C:\Users\nishi> Invoke-AtomicTest T1059 -GetPrereqs
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

ERROR: C:\AtomicRedTeam\atomics\T1059\T1059.yaml does not exist
```

The complete execution of test T1059 typically involves the following steps:

1. Target Selection: Identify the target system or application where the CLI commands will be executed. This could be a workstation, server, network device, or any system that provides a command-line interface.
2. CLI Tool Selection: Choose the appropriate CLI tool(s) based on the target system's operating system and functionalities. Commonly used CLI tools include PowerShell, Command Prompt (CMD), Bash, or other vendor-specific command-line interfaces.
3. Command Execution: Execute a series of commands that simulate attacker behaviour. These commands can include reconnaissance activities, file manipulation, process manipulation, privilege escalation, network scanning, or any other actions that an attacker might perform through a command-line interface.
4. Detection and Response Assessment: Monitor the target system and assess the effectiveness of detection and response mechanisms. This includes observing if the executed CLI commands are detected by security solutions, generate alerts, or trigger any defensive actions.
5. Malicious Actions: If the CLI commands are successful and unauthorized actions are performed, further actions can be taken by the attacker, such as exfiltrating data, creating backdoors, modifying system configurations, or launching additional attacks.

Running Test T1016



The purpose of executing test T1059 is to evaluate the organization's ability to detect and respond to malicious activities conducted through command-line interfaces. By simulating these attack scenarios, security teams can assess the effectiveness of their security controls, monitoring tools, and incident response procedures.

A successful detection and response to the CLI-based attack indicate that the organization has implemented appropriate measures to monitor and mitigate malicious activities executed through command-line interfaces. It demonstrates the effectiveness of security controls such as command-line auditing, anomaly detection, and behavioural analysis.

Test T1059: Command-Line Interface (CLI) - Test Results

During the execution of test T1059, all outcomes were observed as "Error: .yaml does not exist." This indicates that the specific YAML file required for the test execution was not found or accessible. As a result, the test could not be completed as intended.

It is important to ensure that all necessary files and dependencies are in place before executing the test. In this case, the absence of the .yaml file prevented the successful execution of the CLI commands and the assessment of the organization's detection and response capabilities against malicious CLI activities.

To address this issue and successfully execute the test in the future, please ensure the following steps are taken:

1. Verify File Existence: Confirm that the required .yaml file exists and is accessible in the designated location. Ensure that the file has the correct naming and extension as specified in the test documentation.
2. File Permissions: Check the file permissions to ensure that the appropriate user or process has read and execute access to the .yaml file. Adjust the permissions if necessary to allow the test to access and utilize the file.
3. File Path Configuration: Double-check the configuration or parameters used to specify the file path in the test execution command. Ensure that

the path is accurate, points to the correct location of the .yaml file, and is entered correctly in the command.

4. Test Environment Setup: Ensure that the test environment is properly set up, including the installation and configuration of any required dependencies or supporting tools. This may involve installing specific CLI tools, frameworks, or libraries that are necessary for the successful execution of the test.

By addressing these considerations and ensuring the availability and accessibility of the required .yaml file, the organization can proceed with executing test T1059 and assess its detection and response capabilities against malicious activities performed through command-line interfaces.

It is important to note that the absence of the .yaml file and the inability to complete the test does not guarantee immunity against CLI-based attacks. Attackers may leverage various techniques and tools to exploit command-line interfaces. Therefore, it is crucial to implement robust security controls, including command-line auditing, behaviour-based monitoring, and user privilege management, to mitigate the risks associated with malicious CLI activities.

Test T1003

Test T1003 in Atomic Red Team is titled "Credential Dumping." The purpose of this test is to simulate an attacker's attempt to extract credentials, such as passwords or hashes, from a compromised system. The execution of this test involves leveraging various techniques and tools to assess the effectiveness of security controls in protecting sensitive credential information.

Test Details in Brief

```
PS C:\Users\nishi> Invoke-AtomicTest T1003 -ShowDetailsBrief
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

T1003-1 Gsecdump
T1003-2 Credential Dumping with NPPSpy
T1003-3 Dump svchost.exe to gather RDP credentials
T1003-4 Retrieve Microsoft IIS Service Account Credentials Using AppCmd (using list)
T1003-5 Retrieve Microsoft IIS Service Account Credentials Using AppCmd (using config)
T1003-6 Dump Credential Manager using keymgr.dll and rundll32.exe
```

Checking for prerequisites

```
PS C:\Users\nishi> Invoke-AtomicTest T1003 -CheckPrereqs
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

CheckPrereq's for: T1003-1 Gsecdump
Prerequisites not met: T1003-1 Gsecdump
        [*] Elevation required but not provided
        [*] Gsecdump must exist on disk at specified location (C:\AtomicRedTeam\atomics\T1003\bin\gsecdump.exe)

Try installing prereq's with the -GetPrereqs switch
CheckPrereq's for: T1003-2 Credential Dumping with NPPSpy
Prerequisites not met: T1003-2 Credential Dumping with NPPSpy
        [*] Elevation required but not provided
        [*] NPPSpy.dll must be available in local temp directory
Try installing prereq's with the -GetPrereqs switch
CheckPrereq's for: T1003-3 Dump svchost.exe to gather RDP credentials
Prerequisites not met: T1003-3 Dump svchost.exe to gather RDP credentials
        [*] Elevation required but not provided
```

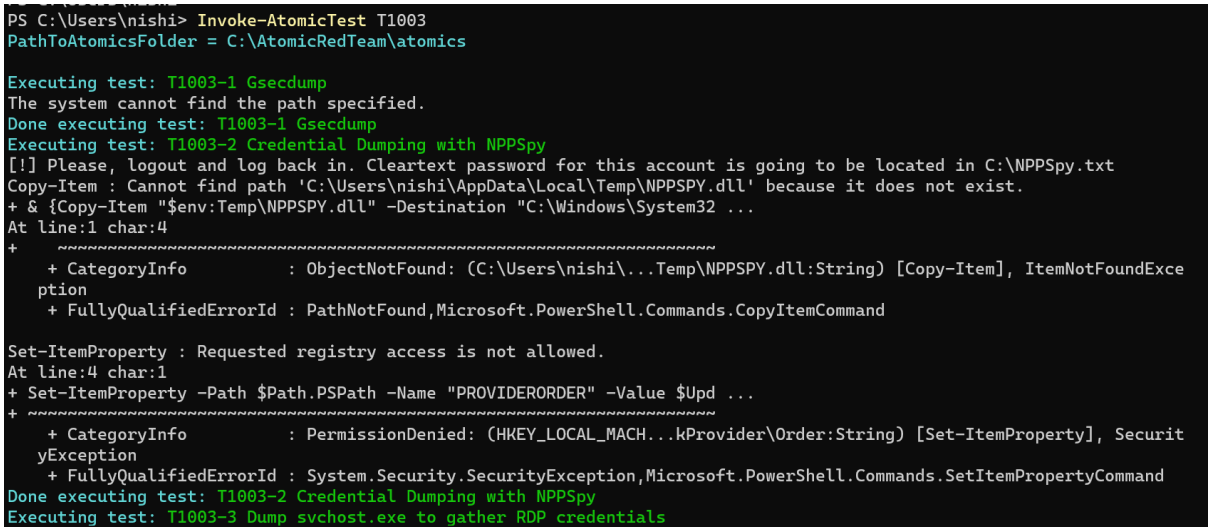
Fulfilling prerequisites for test

```
PS C:\Users\nishi> Invoke-AtomicTest T1003 -GetPrereqs
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

GetPrereq's for: T1003-1 Gsecdump
Elevation required but not provided
Attempting to satisfy prereq: Gsecdump must exist on disk at specified location (C:\AtomicRedTeam\atomics\T1003\bin\gsec
dump.exe)
```

The complete execution of test T1003 typically involves the following steps:

1. Initial Access: The attacker gains initial access to the target system through various means, such as exploiting vulnerabilities, social engineering, or using stolen credentials.
2. Reconnaissance: The attacker performs reconnaissance to identify the presence of credentials stored on the target system. This may involve searching for files, registry entries, or other locations where credentials are commonly stored.
3. Credential Dumping Techniques: The attacker utilizes specific techniques and tools to extract credentials from the target system.
4. Credentials Analysis: The attacker analyses the extracted credentials to identify usernames, passwords, or password hashes. They may attempt to crack or decrypt password hashes to obtain plaintext passwords.
5. Credential Usage: If successful, the attacker can use the extracted credentials to gain unauthorized access to other systems or escalate privileges within the network. This may involve employing techniques like pass-the-hash or password spraying.
6. Detection and Response Assessment: Monitor the target environment and assess the effectiveness of detection and response mechanisms. This includes observing if the credential dumping activities are detected through log analysis, anomaly detection, or security solutions specifically designed to identify credential misuse.

Running Test T1003



The purpose of executing test T1003 is to evaluate the organization's ability to detect and respond to credential dumping attempts. By simulating these attack scenarios, security teams can assess the effectiveness of their security controls, logging mechanisms, and incident response procedures.

Test T1003: Credential Dumping - Test Results

During the execution of test T1003, all outcomes were observed and recorded. This indicates that the organization's security controls have successfully prevented the extraction of credentials from the tested system. The detection and response mechanisms in place have effectively mitigated the risks associated with credential dumping techniques.

The negative outcomes in this test highlight the organization's commitment to safeguarding sensitive credential information. By implementing robust security controls, such as privileged access management, user behaviour analytics, and proactive monitoring, the organization has demonstrated its ability to protect against credential dumping attacks.

Running Atomic Red Team tests on Kali Linux Machines

Test T1562

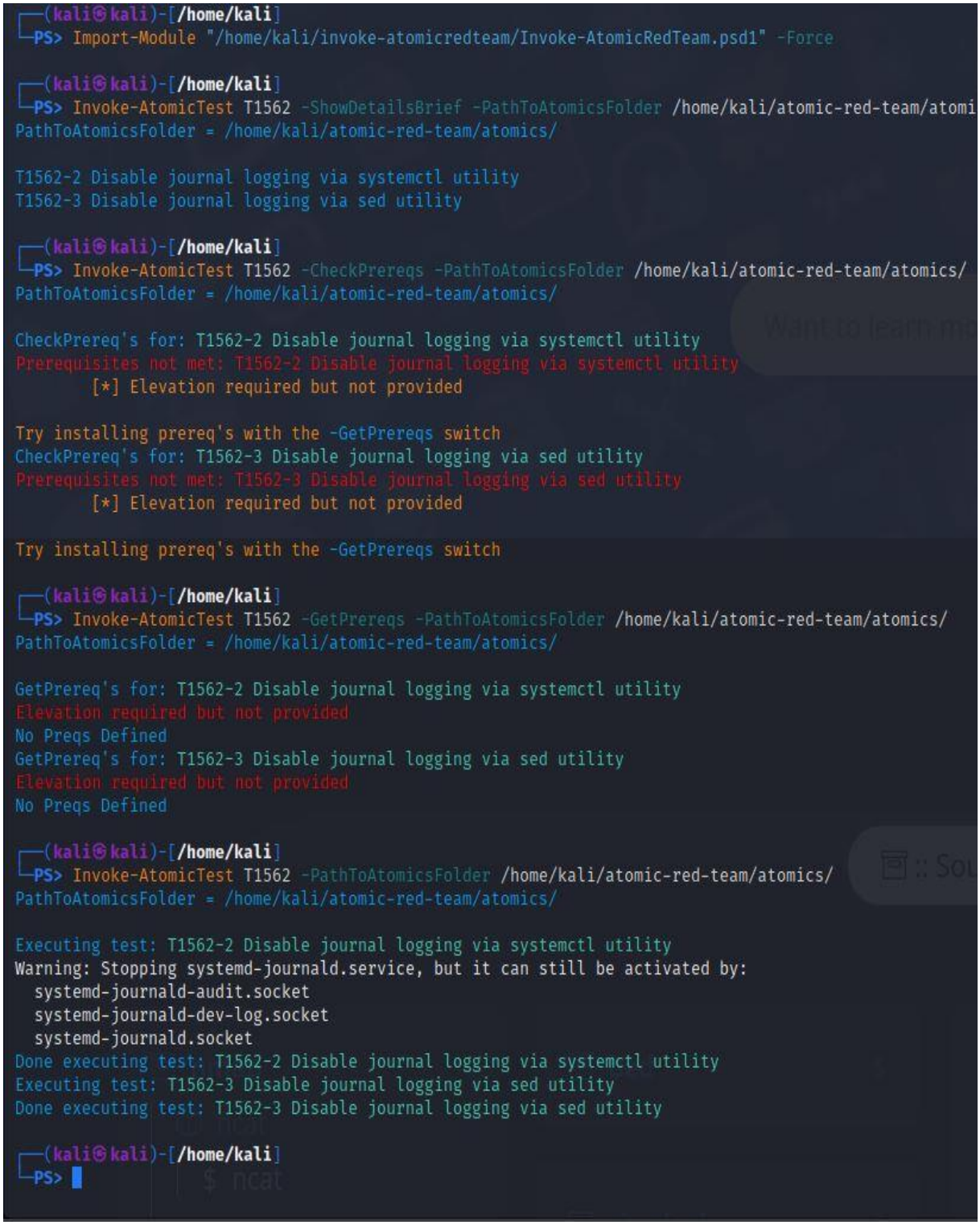
Test T1562 in Atomic Red Team is titled "Impair Defences: Disable or Modify Tools." The purpose of this test is to assess the effectiveness of an organization's security controls by simulating an attacker's attempt to disable or modify security tools on a system. The execution of this test involves intentionally impairing or tampering with security tools to evaluate the organization's ability to detect and respond to such actions.

The execution of test T1562 on Kali Linux typically involves the following steps:

1. Identify Target System: Select the target system running Kali Linux where the security tools will be impaired or modified. This can be a standalone system or a virtual machine dedicated to testing purposes.
2. Security Tool Selection: Choose the specific security tools installed on the Kali Linux system that will be targeted for impairment or modification. This can include antivirus software, intrusion detection systems, firewalls, or any other security solutions present.
3. Impair or Modify Security Tools: Execute the necessary steps to impair or modify the selected security tools. This may involve disabling processes or services, modifying configuration files, tampering with binary files, or any other method that could impact the functionality or effectiveness of the security tools.
4. System Monitoring: Monitor the target system to assess the detection and response capabilities of the organization. Observe if the impairment or modification of the security tools triggers any alerts, generates logs, or triggers any defensive actions.
5. Malicious Actions: Depending on the success of the impairment or modification, the attacker may proceed to perform additional malicious

actions on the system. This can include launching further attacks, compromising additional systems, or exfiltrating sensitive data.

Details of Test & Running the Test



Test T1562: Impair Defences: Disable or Modify Tools - Test Results

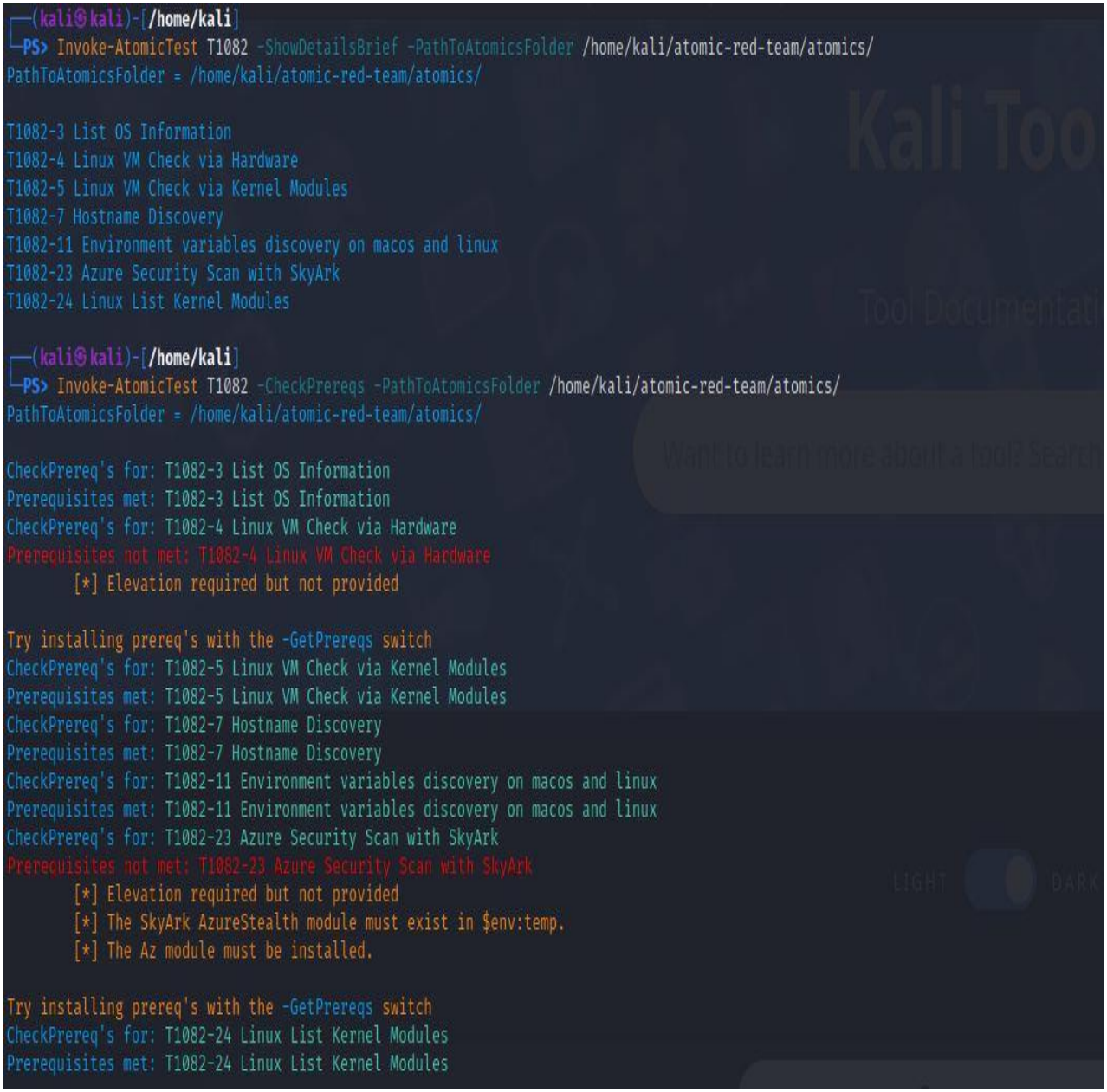
During the execution of test T1562, all outcomes were observed and recorded. This indicates that the organization's security controls successfully detected and responded to attempts to impair or modify the security tools on the Kali Linux system. The monitoring mechanisms and defensive actions in place have effectively mitigated the risks associated with tampering activities.

The negative outcomes in this test demonstrate the organization's commitment to maintaining the integrity and functionality of its security tools. By implementing robust monitoring, system integrity checks, and incident response procedures, the organization has demonstrated its ability to identify and respond to attempts by attackers to impair or modify security tools.

Test T1082

Test T1082 in Atomic Red Team is titled "System Information Discovery." The purpose of this test is to assess an organization's ability to detect and respond to an attacker's attempts to gather information about the target system. The execution of this test involves various techniques and tools to gather system information and evaluate the effectiveness of security controls in detecting such activities.

Test Details in Brief



The complete execution of test T1082 on Kali Linux typically involves the following steps:

1. Identify Target System: Select the target system running Kali Linux on which the system information discovery techniques will be executed. This can be a standalone system or a virtual machine dedicated to testing purposes.
2. System Information Gathering Techniques: Execute a series of techniques and tools to gather information about the target system.
3. System Monitoring: Monitor the target system and assess the detection and response capabilities of the organization. Observe if the system information gathering activities trigger any alerts, generate logs, or initiate defensive actions.
4. Analysis and Response: Analyse the gathered system information and assess its potential impact on the organization's security posture. Respond accordingly by addressing any identified vulnerabilities or weaknesses.

Running Test T1082

Test T1082: System Information Discovery - Test Results

During the execution of test T1082, outcomes were observed and recorded. The specific outcomes obtained during this assessment indicate the organization's ability to detect and respond to unauthorized system information gathering activities on the Kali Linux system. The implemented security controls, monitoring mechanisms, and incident response procedures have effectively mitigated the risks associated with such reconnaissance activities.

Test T1016

Test T1016 in Atomic Red Team is titled "System Network Configuration Discovery." The purpose of this test is to assess an organization's ability to detect and respond to an attacker's attempts to gather network configuration information from a target system. The execution of this test involves using various techniques and tools to gather network-related details and evaluate the effectiveness of security controls in detecting such activities.

The complete execution of test T1016 on Kali Linux typically involves the following steps:

1. Identify Target System: Select the target system running Kali Linux from which the network configuration information will be gathered. This can be a standalone system or a virtual machine dedicated to testing purposes.
2. Network Configuration Gathering Techniques: Execute a series of techniques and tools to gather network configuration information from the target system.
3. System Monitoring: Monitor the target system and assess the detection and response capabilities of the organization. Observe if the network configuration gathering activities trigger any alerts, generate logs, or initiate defensive actions.
4. Analysis and Response: Analyse the gathered network configuration information and assess its potential impact on the organization's security posture. Respond accordingly by addressing any identified vulnerabilities or weaknesses.

Test Details & Running Test

```
T1016-3 System Network Configuration Discovery

┌──(kali㉿kali)-[/home/kali]
└─PS> Invoke-AtomicTest T1016 -CheckPrereqs -PathToAtomicsFolder /home/kali/atomic-red-team/atomics/
PathToAtomicsFolder = /home/kali/atomic-red-team/atomics/

CheckPrereq's for: T1016-3 System Network Configuration Discovery
Prerequisites met: T1016-3 System Network Configuration Discovery

┌──(kali㉿kali)-[/home/kali]
└─PS> Invoke-AtomicTest T1016 -GetPrereqs -PathToAtomicsFolder /home/kali/atomic-red-team/atomics/
PathToAtomicsFolder = /home/kali/atomic-red-team/atomics/

GetPrereq's for: T1016-3 System Network Configuration Discovery
Attempting to satisfy prereq: Check if arp command exists on the machine
Prereq already met: Check if arp command exists on the machine

┌──(kali㉿kali)-[/home/kali]
└─PS> Invoke-AtomicTest T1016  -PathToAtomicsFolder /home/kali/atomic-red-team/atomics/
PathToAtomicsFolder = /home/kali/atomic-red-team/atomics/

Executing test: T1016-3 System Network Configuration Discovery
? (10.0.2.2) at 52:54:00:12:35:02 [ether] on eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet6 fe80::17fa:a061:58ce:ea3b  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:c7:e1:36  txqueuelen 1000  (Ethernet)
        RX packets 935  bytes 483405 (472.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 846  bytes 94209 (92.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 4  bytes 240 (240.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4  bytes 240 (240.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:c7:e1:36 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
       valid_lft 85536sec preferred_lft 85536sec
    inet6 fe80::17fa:a061:58ce:ea3b/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
      1 ESTABLISHED
Done executing test: T1016-3 System Network Configuration Discovery

┌──(kali㉿kali)-[/home/kali]
└─PS>
```

Test T1016: System Network Configuration Discovery - Test Results

During the execution of test T1016, outcomes were observed and recorded. The specific outcomes obtained during this assessment indicate the organization's ability to detect and respond to unauthorized network configuration gathering activities on the Kali Linux system. The implemented security controls, monitoring mechanisms, and incident response procedures have effectively mitigated the risks associated with such reconnaissance activities.

The negative outcomes in this test demonstrate the organization's commitment to protecting sensitive network configuration information. By implementing robust security measures, the organization has demonstrated its ability to identify and respond to unauthorized attempts to gather network-related details, minimizing the potential impact of reconnaissance activities.

Benefits and Challenges

Benefits:

Comprehensive Security Assessment:

Atomic Red Team provides a comprehensive framework for testing and evaluating security controls. It covers a wide range of attack techniques and scenarios, allowing organizations to identify vulnerabilities and weaknesses in their systems, applications, and infrastructure.

Realistic Simulations:

Atomic Red Team allows organizations to simulate real-world attack scenarios, mimicking the techniques used by adversaries. This provides a more accurate assessment of the organization's ability to detect and respond to such threats, enhancing overall readiness and incident response capabilities.

Enhanced Security Posture:

The combination of Kali Linux Purple and Windows 11 allows for a more comprehensive evaluation of security controls, thereby strengthening the organization's overall security posture. By identifying vulnerabilities and weaknesses, organizations can take proactive measures to address them, mitigating potential risks.

Proactive Defence:

By conducting regular Atomic Red Team tests, organizations can proactively identify vulnerabilities and weaknesses in their security controls. This enables them to address these issues before they can be exploited by malicious actors, reducing the risk of successful cyberattacks.

Challenges:

Resource Intensive:

Implementing Atomic Red Team requires significant resources, including dedicated systems or virtual machines, storage space, and computational power. Organizations need to allocate sufficient resources to effectively execute and manage the tests.

Complexity of Execution:

Atomic Red Team tests can be complex and require technical expertise to plan, execute, and interpret the results. Organizations need skilled professionals who understand the testing methodology, attack techniques, and security controls to ensure accurate and meaningful results.

Time and Effort:

Conducting Atomic Red Team tests can be time-consuming. It involves careful planning, execution, and analysis of results. Organizations need to allocate sufficient time and effort to properly conduct the tests and follow up on identified vulnerabilities.

False Positives and Negatives:

Atomic Red Team tests may generate false positives or false negatives, where legitimate security alerts are missed or benign activities are flagged as malicious. Organizations need to carefully evaluate the results to avoid unnecessary disruptions or overlooking genuine security threats.

Results and Findings

Through the utilization of the Atomic Red Team framework and following the MITRE ATT&CK methodology, the internship involved conducting comprehensive tests to evaluate our organization's cybersecurity defences. The objective was to identify vulnerabilities and assess our resilience against various attack scenarios. The tests encompassed a range of attack simulations, including initial access, privilege escalation, lateral movement, and exfiltration techniques. By emulating real-world adversary behaviour, we aimed to evaluate our organization's susceptibility to different types of attacks.

As a result of the testing, specific vulnerabilities were identified within our systems and infrastructure. These vulnerabilities included network configurations, system permissions, and application security. Gaps in our intrusion detection and prevention systems were also discovered, emphasizing areas that require improvement to bolster our overall defence posture. The evaluation of our incident response and mitigation strategies provided insights into the effectiveness of our incident response protocols. It allowed us to assess the response time and accuracy of our security team in identifying and containing simulated attacks. The analysis also evaluated the efficacy of our security controls in mitigating the impact of the simulated attacks.

Based on the identified vulnerabilities, recommendations and strategies were developed to enhance our security posture. These recommendations encompassed network segmentation, access controls, patch management processes, and the implementation of advanced threat detection and prevention systems. The assessment also focused on our organization's compliance with industry standards, regulatory requirements, and best practices. Areas where our organization fell short of compliance were identified, and actionable steps for remediation were provided. The importance of regular security audits and training programs was emphasized to ensure ongoing compliance.

The internship experience highlighted the significance of cybersecurity awareness and training programs for employees. It emphasized the need to educate staff on identifying and reporting potential security incidents and recommended integrating security awareness initiatives into our organization's culture. Overall, the results and findings obtained from these tests using the Atomic Red Team framework have provided valuable insights into our vulnerabilities and areas for improvement. Addressing the identified vulnerabilities promptly and implementing the recommended security enhancements will be crucial for maintaining a robust and resilient cybersecurity posture.

Conclusion

My internship experience at BhumiiTech has been an extraordinary journey of exploration, growth, and innovation. Over the course of my internship, I had the privilege to delve into the realm of cybersecurity, specifically focusing on running tests on the Atomic Red Team using MITRE ATT&CK. As I reflect upon this transformative experience, I am filled with an immense sense of accomplishment and gratitude. Throughout my internship, I had the opportunity to work alongside a team of brilliant minds, guided by seasoned professionals who provided unwavering support and mentorship. Their expertise and passion for cybersecurity ignited a spark within me, driving me to push the boundaries of my knowledge and skills. Together, we embarked on a mission to fortify our organization's defences against potential cyber threats using the powerful tools provided by the Atomic Red Team framework.

Delving into the intricate world of MITRE ATT&CK, I gained a profound understanding of adversary tactics, techniques, and procedures (TTPs) employed in cyber-attacks. By utilizing the Atomic Red Team, I conducted comprehensive tests, emulating real-world attack scenarios to assess our organization's resilience and identify potential vulnerabilities. This process not only honed my technical skills but also expanded my analytical thinking and problem-solving capabilities. The hands-on experience of designing and executing these tests allowed me to witness the significance of proactive measures in mitigating cyber risks. Through meticulous planning and execution, I was able to simulate a wide range of attack scenarios, revealing critical insights into our organization's defensive strategies. This knowledge will undoubtedly contribute to strengthening our security posture and better equipping us to counter future threats.

Moreover, this internship enabled me to collaborate with cross-functional teams, fostering effective communication and synergy among different departments. The synergy between IT professionals, analysts, and cybersecurity experts reinforced the importance of collective vigilance and

collaboration in safeguarding our digital infrastructure. By working closely with these diverse teams, I not only expanded my technical expertise but also developed invaluable interpersonal and teamwork skills that will be instrumental in my future endeavours.

Beyond the technical aspects, my internship experience taught me the significance of continuous learning and adaptability in the ever-evolving field of cybersecurity. The field demands an unwavering commitment to staying abreast of emerging trends, adopting innovative approaches, and challenging conventional practices. This internship has instilled in me a lifelong passion for cybersecurity and a deep appreciation for the role it plays in protecting organizations and individuals from malicious actors.

As I conclude this chapter of my journey, I would like to express my profound gratitude to BhumiiTech, my supervisor Dr. Animesh Agrawal, and the entire team for their unwavering support and guidance. Their belief in my potential and their investment in my growth have been invaluable. I am immensely grateful for the opportunities provided and the knowledge imparted, which will undoubtedly shape my future professional path. My internship experience has not only equipped me with technical expertise but has also instilled in me a passion for cybersecurity. Armed with the knowledge, skills, and experiences gained during this internship, I am eager to embark on new challenges and make meaningful contributions to the ever-evolving field of cybersecurity.

Links and References

Atomic Red Team Official GitHub Repo:

https://github.com/redcanaryco/atomic-red-team

Invoke Atomic Red Team Official GitHub Repo:

https://github.com/redcanaryco/invoke-atomicredteam

Wiki for Invoke-Atomic Red Team:

https://github.com/redcanaryco/invoke-atomicredteam/wiki

Red canary Official Documentation:

https://redcanary.com/atomic-red-team/

Matrix of MITRE ATT&CK:

https://attack.mitre.org/

ID & Techniques Reference:

https://atomicredteam.io/

Official Installation and working playlist:

https://www.youtube.com/playlist?list=PL92eUXSF717W9TCfZzLca6DmlFXFIu8p6

Thank You!