Emulating the Highly Sophisticated North Korean Adversary Lazarus Group – Part 1

By Francis Guibernau

Published: 2023-01-05 · Archived: 2026-04-05 22:05:27 UTC

Lazarus Group, also known as Hidden Cobra, is a state-sponsored adversary attributed to the Reconnaissance General Bureau (RGB) of the Democratic People's Republic of Korea (DPRK) which has been active since at least 2009. The Lazarus Group is composed of at least two subgroups, both known as Andariel and BlueNoroff, and has notable overlaps with the adversaries known as APT37 and Kimsuky.

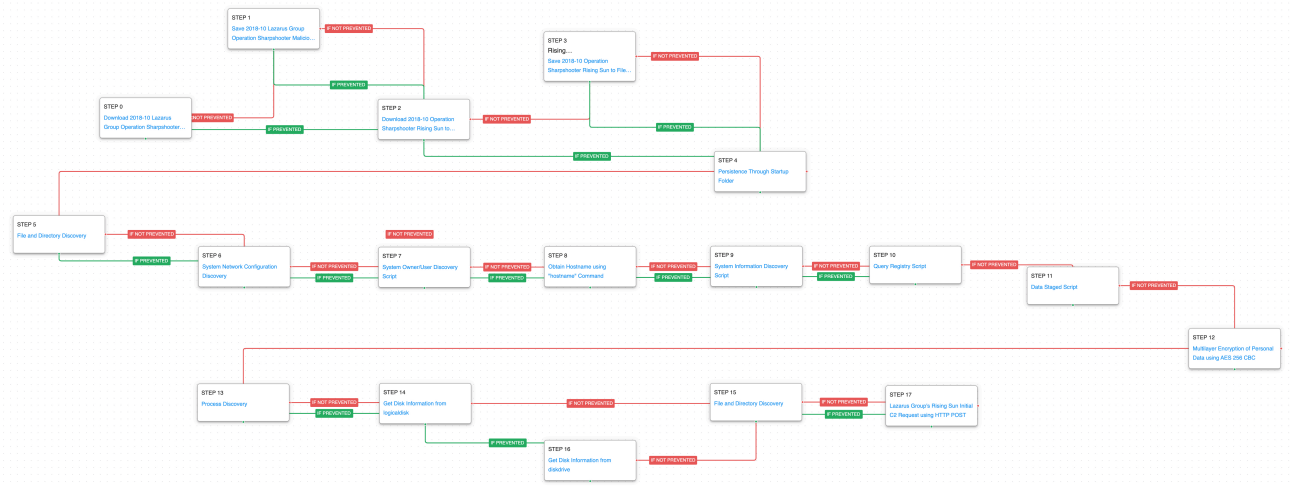
Lazarus Group's main motivations are theft of proprietary information, espionage, sabotage, and destruction. The group first came to media attention in 2013, following a series of coordinated attacks against South Korean media and financial entities using the wiper known as DarkSeoul.

Their most notorious campaign occurred in November 2014 when the Lazarus Group conducted a large-scale destructive attack against Sony Pictures Entertainment (SPE), which was notable due to the substantial penetration through the network, the large amount of exfiltrated data, and the use of a wiper to erase all forensic evidence.

AttackIQ has released six new attack graphs emulating the actor's historical campaigns to help customers validate their security controls and their ability to defend against this group. Validating your security program performance against these behaviors is vital to reducing risk. By using these new attack graphs in the AttackIQ Security Optimization Platform, security teams will be able to:

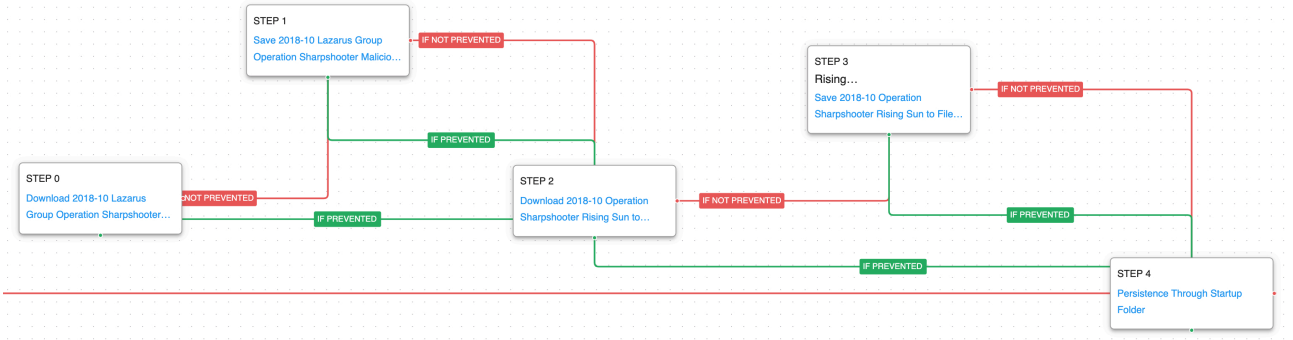
- Evaluate security control performance against the top North Korean threat actor who has targeted all regions and sectors.
- Assess their security posture against a threat actor who is not afraid to commit destructive actions.
- Continuously validate detection and prevention pipelines against the techniques shared amongst many of the North Korean adversaries.

Lazarus Group – 2018-12 – Operation Sharpshooter



(Click for Larger)

The first attack graph is based on Operation Sharpshooter reported by McAfee in December 2018. Operation Sharpshooter took place between October and November 2018 against more than 80 organizations worldwide, predominantly those located in the United States. During this attack, the adversary focused on targeting multiple sectors, specifically those involved in finance, energy, and defense.

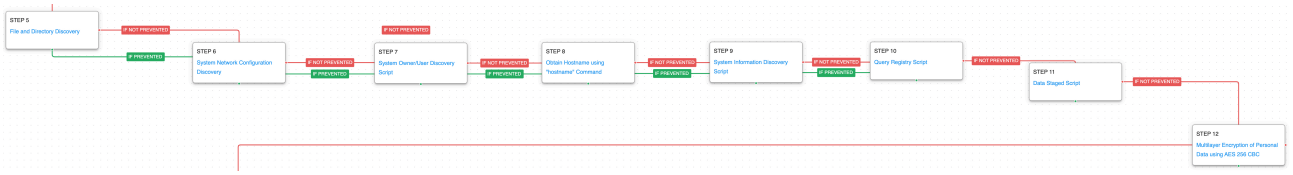


(Click for Larger)

The attack graph begins with the downloading and saving of the malicious Office Document used for the deployment of the Rising Sun implant, which obtains persistence through the Startup folder.

Ingress Tool Transfer (T1105): This scenario downloads to memory and saves to disk in two separate scenarios to test network and endpoint controls and their ability to prevent the delivery of known malicious samples of Lazarus malware. These scenarios are used for each stage of the malware delivered in these attacks.

Logon Autostart Execution: Startup Folder (T1547.001): The Startup folder is a directory associated with the Windows Start Menu that can be used to launch a process at Windows logon. This scenario creates a binary file in this directory that would execute at next logon for users.



(Click for Larger)

During the second stage, the graph seeks to collect information from various sources about the system environment prior to the exfiltration of the encrypted collected data.

File and Directory Discovery (T1083): This scenario uses the native `dir` command to find files of interest and output to a temporary file.

System Network Configuration Discovery (T1016): Native Window's commands like `route` , `ipconfig` , and `net use` are executed to collect details about the infected host and network shares.

System Owner / User Discovery (T1033): Living off the land by running `whoami` and `users` to gain details about the currently available accounts and permission groups.

System Information Discovery (T1082): The native `hostname` and `systeminfo` commands are used to get the infected host's computer name and basic details about the system.

Query Registry (T1012): The `HKCU\Software\Microsoft\Windows\CurrentVersion` registry key contains information about Windows properties for the user account logged into the system.

Data Staged: Local Data Staging (T1074.001): Files are collected and stored in a temporary directory so they can be exfiltrated later.

Exfiltration Over Symmetric Encrypted Non-C2 Protocol (T1048.001): Files are first encrypted using AES encryption and then transmitted over HTTPS using POST requests.

(Click for Larger)

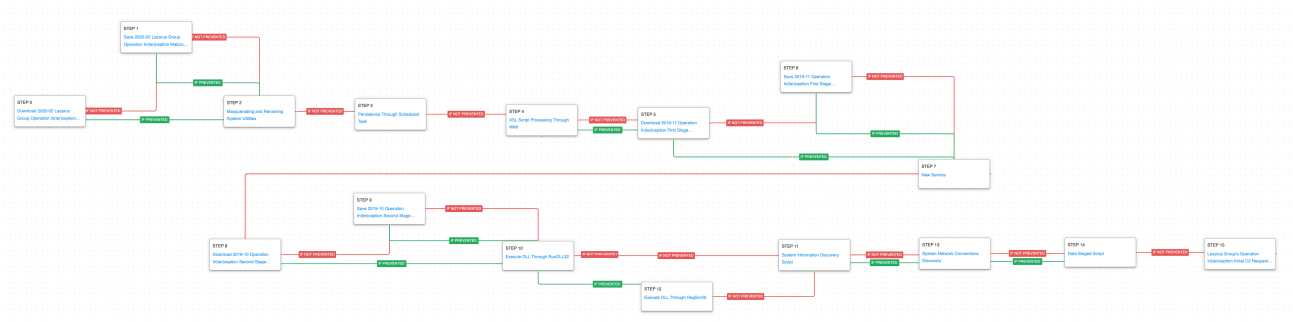
In the last stage, the implant seeks to obtain in-depth information about the files contained in the system and peripheral devices, finalizing with the communication to the adversary's infrastructure.

Process Discovery (T1057): `tasklist` is executed as a command process and the results are saved to a temporary location.

System Information Discovery (T1082): The native Windows commands `logicaldisk` and `diskdrive` are executed to collect information about the installed disks, including caption, description, drive type, provider name, and size.

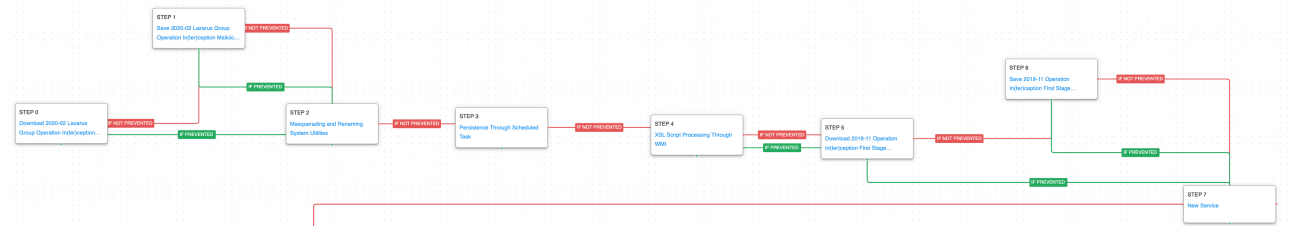
Application Layer Protocol: Web Protocols (T1071.001): This scenario emulates the HTTP requests made by the Rising Sun backdoor by making an HTTP POST to an AttackIQ server that mimics the URL format and data sent by a real infection.

Lazarus Group – 2020-06 – Operation In(ter)ception



(Click for Larger)

The second attack graph is based on the Operation In(ter)ception report by ESET published in June 2020. Operation In(ter)ception was a campaign conducted against military and aerospace organizations in Europe and the Middle East. The actor used social engineering via LinkedIn, hiding behind the ruse of attractive, but bogus, job offers.



(Click for Larger)

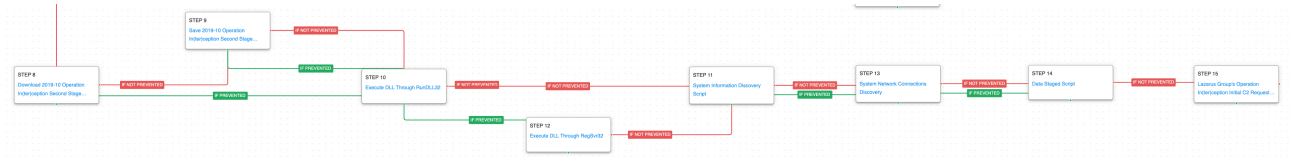
The attack graph begins with the downloading and saving of a malicious LNK file, which seeks to evade defenses by masquerading as a legitimate process when executed. Next, the graph seeks to obtain persistence through a scheduled task and executes a remote XSL script. The first-stage downloader is downloaded, saved, and executed as a new service.

Masquerading: Match Legitimate Name or Location (T1036.005): A copy of the legitimate `wmic.exe` binary is placed in a temporary directory. The file is renamed to `ncv.exe` and executed to get details on the OS version.

Scheduled Task/Job: Scheduled Task (T1053.005): This scenario creates a new scheduled task using the `schtasks` utility with the name `HPSync` that was observed being used in these attacks.

XSL Script Processing (T1220): `wmic.exe` is executed and passed a URL as a command line argument that forces the binary to download and execute an XSL file that contains malicious JavaScript code.

Windows Service (T1543.003): Use the native `sc` command line tool to create a new service that will executed at reboot.



(Click for Larger)

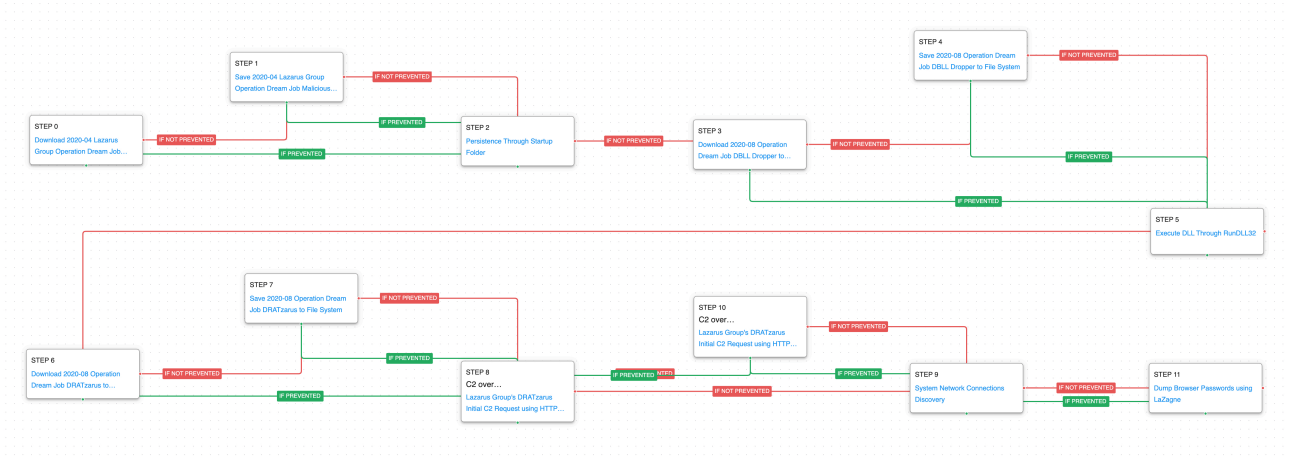
The second stage is downloaded and executed using RunDLL32, and in case it is prevented, it will alternatively be executed through RegSvr32. Finally, the graph will seek to obtain information from the environment exfiltrate to adversary infrastructure.

System Binary Proxy Execution: Rundll32 (T1218.011): `RunDll32` is a native system utility that can be used to execute DLL files and call a specific export inside the file. This scenario executes `RunDll32` with an AttackIQ DLL and calls an export to mimic previously reported malicious activity.

System Binary Proxy Execution: Regsvr32 (T1218.010): `RegSvr32` is a native Windows utility that threat actors can use to register Common Object Model (COM) DLLs. This functionality allows an actor to deploy a malicious DLL and have a native Windows tool execute the code as the parent process. This scenario executes `RegSvr32` with an AttackIQ binary.

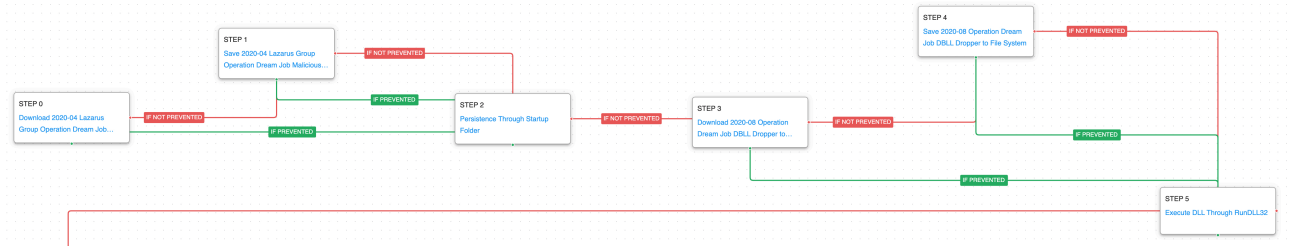
System Network Connections Discovery (T1049): The native Windows command line tool `netstat` is used to collect active connections and any listening services running on the host.

Lazarus Group – 2020-08 – Operation Dream Job (ClearSky)



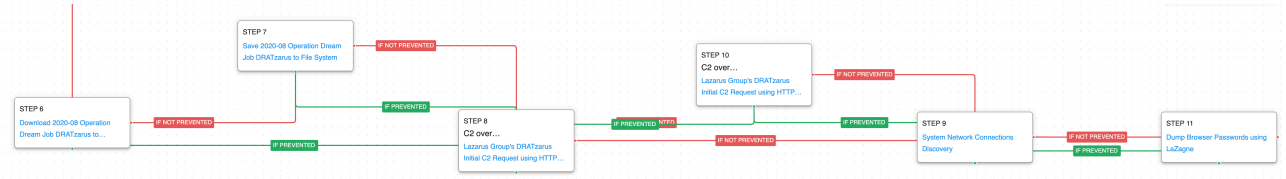
(Click for Larger)

This attack graph emulates the first iteration of the Operation Dream Job, reported by ClearSky in August 2020. Operation Dream Job was a cyberattack carried out against multiple individuals worldwide from early 2020 to mid-2022. The actors used social media to phish victims in the defense sector and various government organizations leveraging fake job offers from prominent defense and aerospace companies.



(Click for Larger)

The attack graph begins with the downloading and saving of a malicious Office document, which obtains persistence through the Startup folder, using a dropped LNK shortcut file. Subsequently, the DBLL Loader is downloaded and saved to the system, which is executed through RunDLL32 and to complete the deployment of the final payload known as DRATzarus.

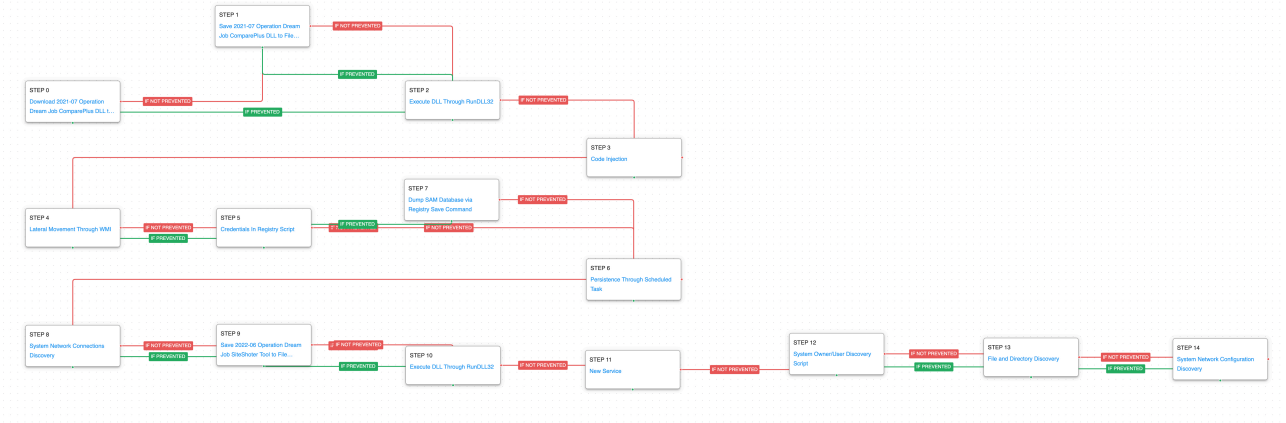


(Click for Larger)

In the last stage, after downloading and saving DRATzarus to the system, the attack graph recreates the communications captured between the malware sample and the adversary's infrastructure, ending with the collection of the network connections available in the system and obtaining credentials from the browser through LaZagne.

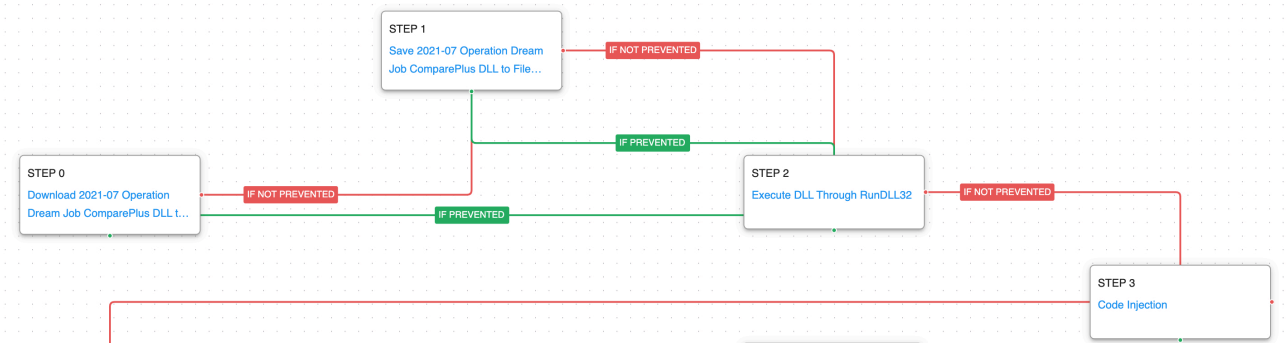
OS Credential Dumping (T1003): This scenario uses the open-source tool LaZagne to dump credentials available on the host including the saved browser passwords.

Lazarus Group – 2022-04 – Operation Dream Job (Symantec)



(Click for Larger)

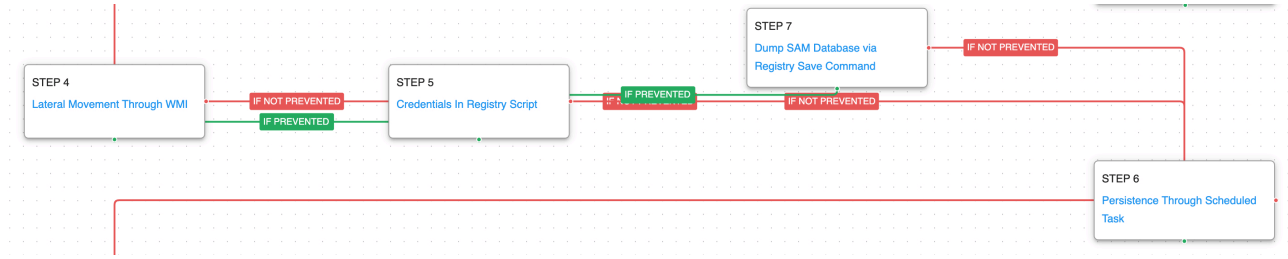
The next attack graph continues to emulate the second iteration of the Dream Job operation, this time from activity reported by Symantec in April 2022.



(Click for Larger)

The attack graph starts by downloading and saving a trojanized version of the ComparePlus plugin DLL, which is a Notepad++ plugin used to compare two files and show differences side by side. After being executed via RunDLL32, the attack graph will seek to inject code into an active process.

Process Injection (T1055): This scenario injects a DLL file into another running process and validates if a canary file can be created.



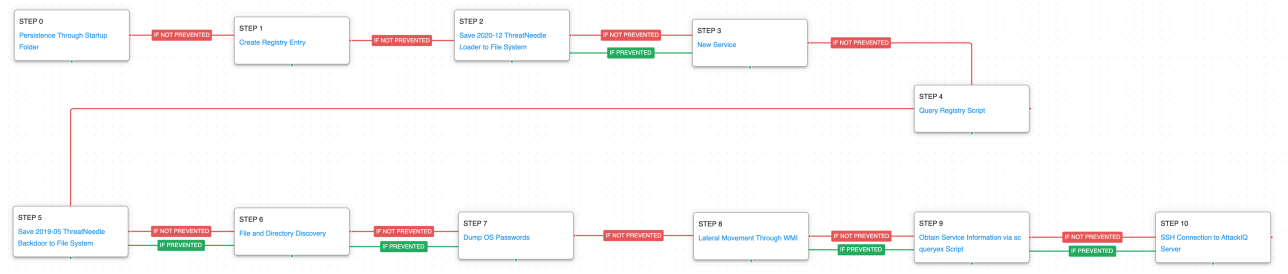
(Click for Larger)

Subsequently, it will attempt to perform lateral movement by using Windows Management Instrumentation (WMI), continuing with obtaining credentials through the registry or by dumping of the SAM database, and finishing with obtaining persistence through a scheduled task.

Windows Management Instrumentation (T1047): This scenario uses `wmic` commands to execute commands on a remote target.

Unsecured Credentials: Credentials in Registry (T1552.002): A PowerShell script is executed that searches for `DefaultPassword` and `AltDefaultPassword` registry keys that contain hard-coded credentials.

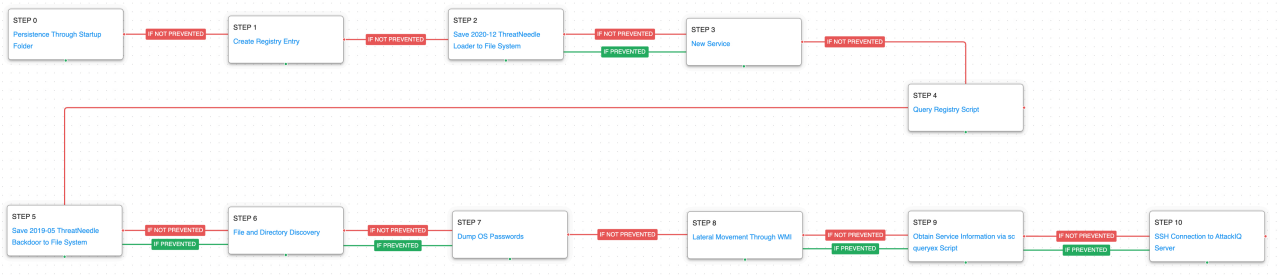
OS Credential Dumping: Security Account Manager (T1003.002): The built-in `reg save` command is executed to dump the Windows `SAM` hive.



(Click for Larger)

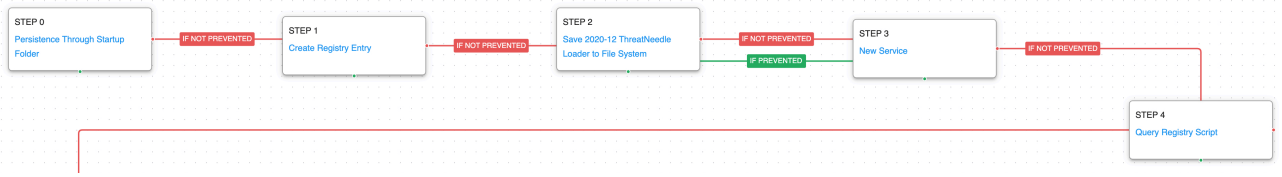
In the last stage, the attack graph will try to obtain information about the active network connections and drop their tool known as SiteShooter. Finally, it will collect information about the infected environment, ending up obtaining the system network configuration.

Lazarus Group – 2021-01 – ThreatNeedle Campaign



(Click for Larger)

The ThreatNeedle campaign was a cyberattack against security researchers and the defense industry from mid-2020 to early 2021 reported by Kaspersky. During this attack, Lazarus made use of the malware family known as ThreatNeedle, which is based on the Manuscrypt family. The group leveraged COVID-19-themed emails, personalizing them with personal information obtained during an initial reconnaissance effort.



(Click for Larger)

The attack graph begins with obtaining persistence through the Startup folder and immediately continues with the deployment of the ThreatNeedle Loader, which seeks to create a new service to load the ThreatNeedle backdoor.

Modify Registry (T1112): This scenario sets the same registry key used by the actor by calling the `New-ItemProperty` cmdlet.

(Click for Larger)

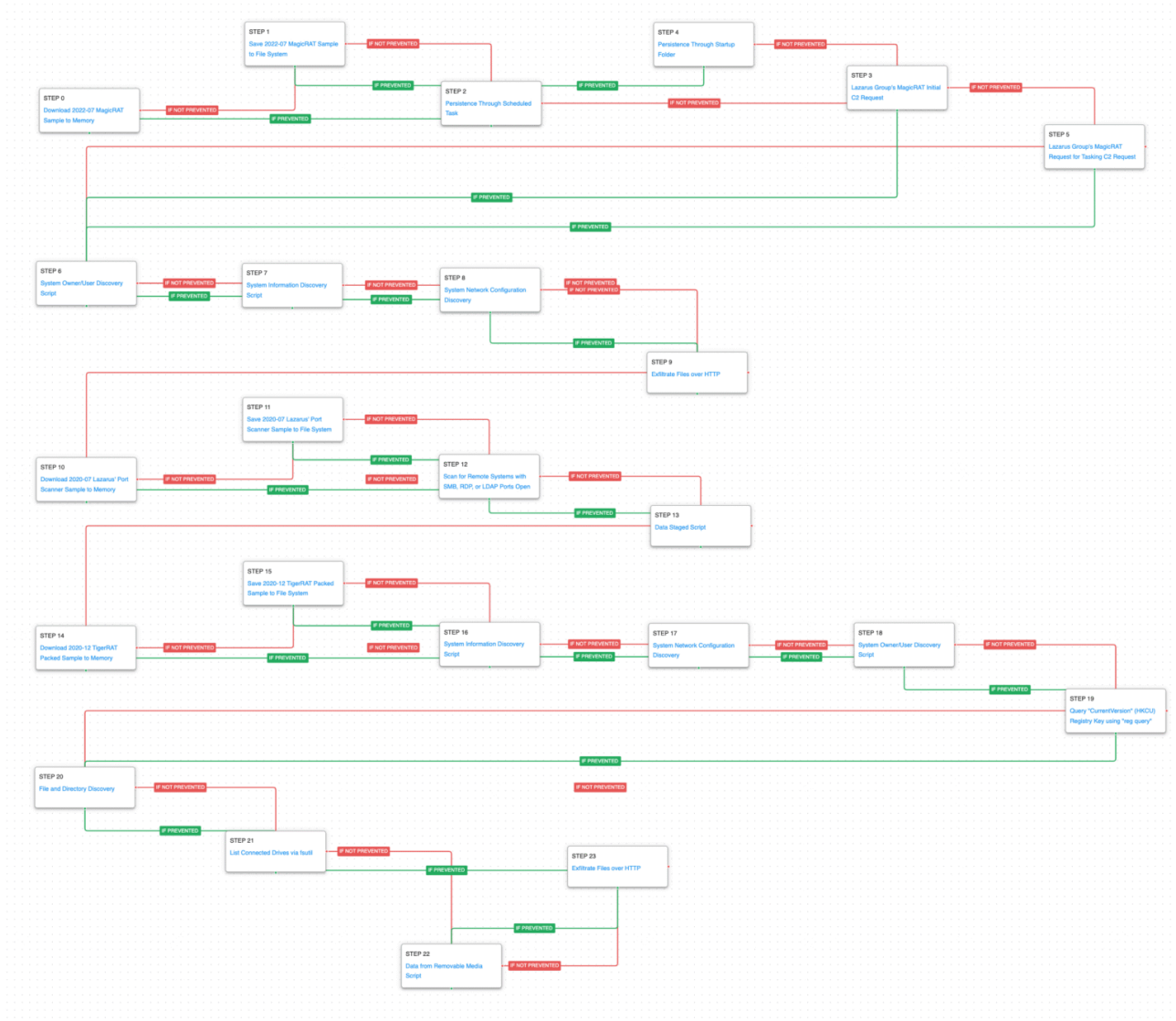
In the last stage, the attack graph will seek to collect system information, obtain credentials, and move laterally through the victim network, ending with tunneling an SSH connection to external infrastructure.

OS Credential Dumping (T1003): This scenario uses the open-source tool MimiKatz to dump all possible credentials available on the host.

System Service Discovery (T1007): Microsoft's native `sc` utility is executed to query a list of all running services.

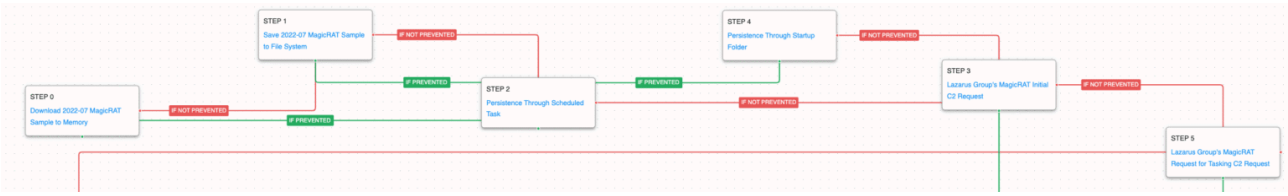
Protocol Tunneling (T1572): This scenario tests security controls responsible for blocking outbound SSH connections to external servers.

Lazarus Group – 2022-07 – MagicRAT + TigerRAT Campaign



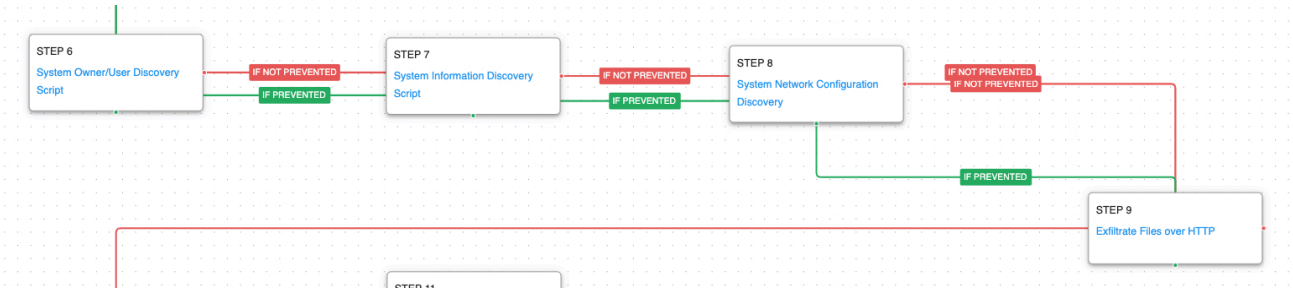
(Click for Larger)

The sixth and final attack graph seeks to emulate the activity reported by Cisco Talos in September 2022. During this activity, researchers observed the compromise of victims with a new Remote Access Trojan they named MagicRAT delivered by exploiting publicly exposed VMWare Horizon platforms. During this activity, Lazarus Group additionally used TigerRAT, a malware previously attributed to the Andariel adversary during Operation ByteTiger in September 2021.



(Click for Larger)

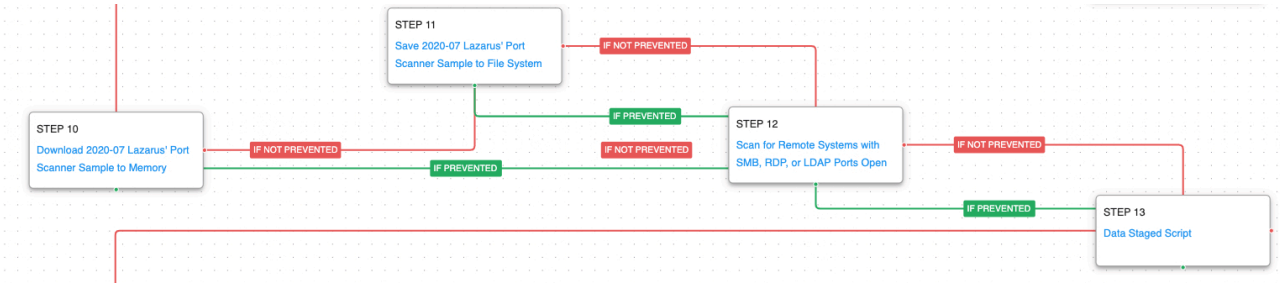
This attack graph starts with the deployment of MagicRAT immediately after the exploitation of vulnerabilities present in VMWare Horizon platforms. During this first stage, MagicRAT will seek to obtain persistence using a scheduled task or the Startup folder prior to checking in and registering with the actor's infrastructure.



(Click for Larger)

In the second stage of the attack, the actor completes some basic discovery commands before attempting to quickly exfiltrate some files of interest.

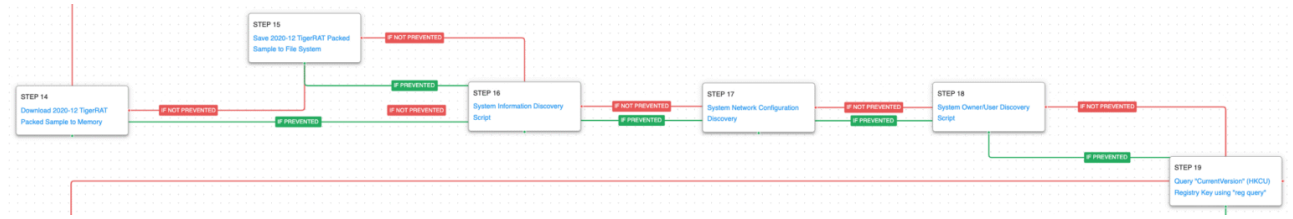
Exfiltration Over C2 Channel (T1041): Files are sent to an AttackIQ controlled server using `HTTP POST` requests.



(Click for Larger)

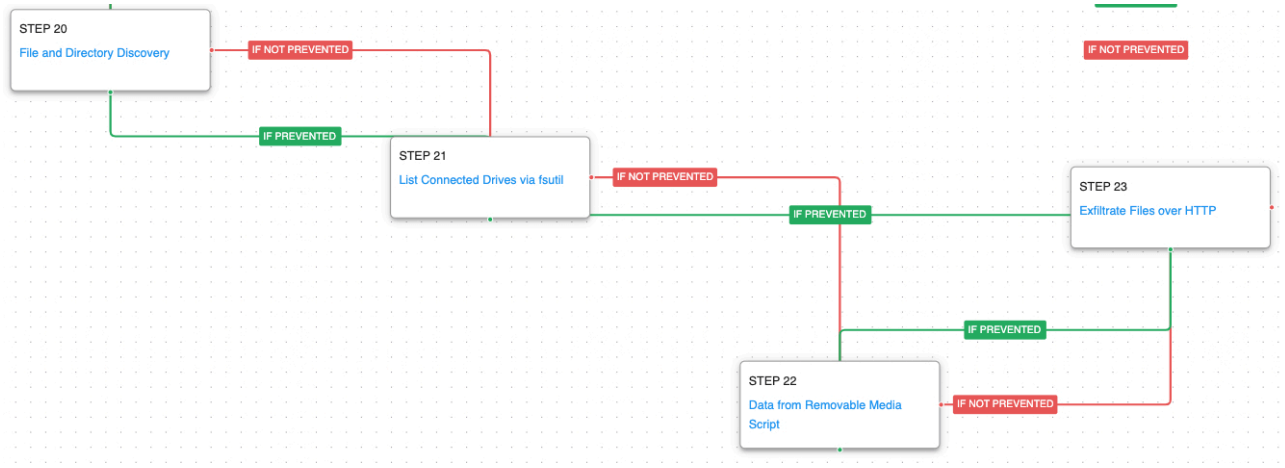
In the next stage, Lazarus Group uses a lightweight port scanner to obtain information about the open ports available on other systems located in the infected host's local network. The actor continues to stage data it discovers for delivery on the next check-in with the command-and-control server.

Network Service Discovery (T1046): This scenario uses `nmap` for scanning hosts that are open on ports `139,389,445,636,3389` that would identify remotely accessible hosts to the attacker.



(Click for Larger)

TigerRAT, a Remote Access Trojan (RAT) previously used during Operation ByteTiger first reported by the KrCERT, is deployed. This RAT will seek to obtain additional information from the environment.



(Click for Larger)

In the final stages of the attack graph, the actor makes one final attempt to look for files of interest and checks for the existing of any additional connected drives and collect data from removable media. All of the data collected during the attack is exfiltrated to the actor's infrastructure.

Data from Removable Media (T1025): The native utility `fsutil` is used to identify any additional hard disks connected to the host. PowerShell is then used to iterate through every removable media device and harvest a list of files.

Detection and Mitigation Opportunities

With so many different techniques being used by threat actors, it can be difficult to know which to prioritize for prevention and detection assessment. AttackIQ recommends first focusing on the following techniques emulated in our scenarios before moving on to the remaining techniques.

The attacks from the Lazarus Group are long drawn-out campaigns that require the actor to collect data and infiltrate the victim over time. One of best ways to limit the actor's ability to live in your network is to scrutinize the persistence mechanisms used by the group.

1. Windows Service (T1543.003)

Actors can create or modify Windows services to repeatedly execute malicious payloads as part of persistence. When Windows boots up, it starts programs or applications called services that perform background system functions.

1a. Detection

The following rules can help identify when that persistence mechanism is being set.

```
Process Name == (Cmd.exe OR Powershell.exe)
Command Line CONTAINS ('sc' AND 'create' AND 'start= "auto"')
```

1b. Mitigation

MITRE ATT&CK has the following mitigation recommendations

- M1047 – Audit
- M1040 – Behavior Prevention on Endpoint
- M1018 – User Account Management

2. Scheduled Task/Job: Scheduled Task (T1053.005)

Adversaries may abuse the Windows Task Scheduler to perform task scheduling for initial or recurring execution of malicious code. There are multiple ways to access the Task Scheduler in Windows. The schtasks utility can be run directly from the command line, or the Task Scheduler can be opened through the GUI within the Administrator Tools section of the Control Panel.

2a. Detection

With an EDR or SIEM Platform, you can detect the following commands being issued to schedule a malicious task

```
Process Name = (“cmd.exe” OR “Powershell.exe”)
Command Line CONTAINS (“schtasks” AND “/CREATE” AND (“cmd” OR “powershell”)
```

2b. Mitigation

MITRE ATT&CK has the following mitigation recommendations

- M1047 – Audit
- M1028 – Operating System Configuration
- M1026 – Privileged Account Management
- M1018 – User Account Management

3. Startup Folder (T1547.001)

Each user profile has their own `StartUp` directory and there is also an additional directory that applies to all folders. Actors may place binaries and scripts in here directly or they can place shortcut LNK files that can point to a file to be executed.

3a. Detection

With an EDR or SIEM Platform, you can identify processes originating from the `StartUp` directory and later add exclusions for known legitimate processes specific to your environment.

```
Command Line CONTAINS (“StartUp”)
```

3b. Mitigation

MITRE ATT&CK does not have any mitigation recommendations as this is an abuse of system features. Auditing and Process Logging are the best options.

Wrap-up

In summary, these attack graphs will evaluate security and incident response processes and support the improvement of your security control posture against one of the most dangerous threat actors in the world today. With data generated from continuous testing and use of these attack graphs, you can focus your teams on achieving key security outcomes, adjust your security controls, and work to elevate your total security program effectiveness against a known and dangerous threat.

AttackIQ stands at the ready to help security teams implement this attack graph and other aspects of the AttackIQ Security Optimization Platform, including through our co-managed security service, AttackIQ Vanguard.
