SEQRITE

A Paradigm Shift in Endpoint Security with next-gen EDR Solution

WHITE PAPER

www.seqrite.com

Author-
Akshay Gaikwad,
Nihar Deshpande

Introduction

Over the last several years, the number of cyber-attacks has increased. In addition, with each passing year, the attacks are becoming more and more sophisticated. These days, many attacks don't even use malware; they rely on living off-the-land techniques, thus reducing the effectiveness of existing protection mechanisms. As a result, the endpoint security products are no longer sufficient to keep organizations protected. Also, your organization already has antivirus and a firewall in place; what more could you need? It's tempting to think that having those tools are enough to help you sleep soundly, knowing your entire network is protected.

Unfortunately, cyber threats continue to grow in complexity and sophistication. It's either you keep up or suffer the consequences. Threat actors can penetrate antivirus and firewalls with a well-constructed phishing email or social engineering trick. That means we already have the tactics used by known cybercriminals in our database. So they will use new tactics to attack our system.

The need is to complement these endpoint security products with another layer that can look for behavioral anomalies and help identify any attacks on the organization. This is where EDR, i.e., Endpoint Detection and Response system, comes into the picture.

Before that, let's see the difference between Traditional Endpoint Antivirus and EDR (Endpoint Detection and Response).

What is Traditional Endpoint Antivirus?

Traditional Endpoint Antivirus software, also known as legacy AV, is the “lowest common denominator” of endpoint security. Antivirus scans an operating system and file system for known malware such as trojans, worms, and ransomware, and upon detecting them, removes them from the system. Legacy AV typically detects malware by comparing binaries to known signatures, performing heuristic analysis to see if running processes or installed software have suspicious properties, and integrity checking, which checks if malware has tampered with existing files on a machine.

Why is an EDR (Endpoint Detection and Response) solution required?

Endpoint detection and response (EDR) solution “records and stores endpoint-system-level behaviors, uses various data analytics techniques to detect suspicious system behavior, provides contextual information, blocks malicious activity, and provides remediation suggestions to restore affected systems.” EDR helps security analysts understand that attackers have already breached an endpoint and help them stop attacks by performing automated or manual actions, such as isolating an endpoint from the network, wiping and reimaging it, or identifying and blocking malicious processes.

It helps in the following ways

Identifying the problems

Let us start with a general flow of ransomware infections that can be correlated with the MITRE ATT&CK framework. There is a general behavioral similarity between most cyber-attacks, although the techniques or the sequence of actions might differ.

The starting point is usually a phishing email, which claims to contain some helpful information. An unsuspecting user may click on the web links embedded in the email or open the attachments, which would lead to the attacker's intended script getting executed on the system. That script would usually download another malicious payload without the user's knowledge and start collecting sensitive information, later exfiltrated to an attacker-controlled server. Afterward, the malware would encrypt the system, and a ransom note displayed.

Let us now dig deeper into the finer details of the attack and look at the various steps involved.

Reconnaissance

The first step of the attack taken by the attackers is gathering information about the target. Techniques which the attackers use for reconnaissance include:

- Active scanning
- Phishing
- Gathering victim-related information

Resource Development

The attackers establish the resources and capabilities necessary to execute a cyberattack in the resource development phase. Some techniques here include:

- Acquiring and/or compromising infrastructure
- Compromising or establishing accounts
- Developing capabilities

Initial Access

In the Initial Access step, attackers attempt to access the IT network and infiltrate the network and end-user systems. Standard techniques to gain a foothold within the network include:

- Drive-by compromise
- Spear phishing
- Exploiting external remote services and weak passwords
- Using compromised accounts and vulnerabilities to execute broader attacks later.

Execution

In this phase, malicious code is executed on the target network. They may compromise built-in scripting settings and interpreters to run custom code for network exploration, stealing data and credentials. Standard target interpreters include:

- PowerShell, Windows Command Shell, and Unix Shell
- Python and JavaScript installations.
- Using compromised accounts and vulnerabilities to execute broader attacks later.

Credential Access

Credential access is the stage when attackers steal account credentials. Standard techniques in this phase include:

- Keylogging
- Brute force
- Password cracking/guessing to access systems and approve rogue accounts within the network.

Persistence

Here, the adversary tries to maintain a foothold and bypass security attempts. Once a code script is executed, the adversaries can prevent defensive actions that would interrupt the attack lifecycle. In addition to evading AV program detection, malware attempts to protect themselves from any activities such as system restart, credential changes, configuration resets, etc. - that might interrupt their intent. Adversaries persist using techniques such as:

- Manipulating accounts
- Modifying SSH authentication keys, authentication packages, services, and registry weaknesses

Lateral Movement

In this stage, the adversaries move laterally across the network environment, diverting between systems and accounts for stealthier operations. The process applies to compromising more legitimate credentials and network and default OS tools. Techniques include:

- Internal spear phishing
- Remote service exploitation
- SSH hijacking

Privilege Escalation

Privilege escalation occurs when the attackers get access to elevated permissions in the network, such as root and admin access privileges. Techniques include:

- Sudo caching
- Bypassing user access controls
- Port monitoring

Collection

Adversaries collect information and sources required to steal and exfiltrate data, including but certainly not limited to emails, keyboard input, databases, and archives.

Defense Evasion

Adversaries bypass detection by disabling or uninstalling security systems and scripts. They masquerade malicious activities under known and trusted processes under the radar, subverting possible defenses. Standard techniques in this phase include:

- Abuse elevation control mechanism
- Elevated execution
- Token impersonation

Command & Control

At this stage, the attackers control the network and systems with various levels of stealth. The systems act upon commands from the adversary and simulate normal network behavior to avoid possible detection.

The attackers communicate the commands using:

- Existing application layer protocols
- Data encoding
- Data obfuscation
- Multi-stage channels

Exfiltration

In this phase, the attackers finally exfiltrate relevant data from the compromised network. The data is usually compressed and encrypted before transferring outside the network.

Common techniques in this phase include:

- Automated exfiltration
- Exfiltration over web services or physical medium

Impact

The attack lifecycle ends with manipulating, disrupting, or destroying compromised systems, network components, accounts, and data. Techniques in this stage can include:

- Account access removal
- Data destruction
- Data encryption and manipulation
- Disk wipes
- Denial of Service attacks on the network
- Resource hijacking

Introducing HawkkHunt (Seqrite's EDR) Solution

Here's a sample of HawkkHunt (Seqrite's EDR solution) illustrating process-flow and chain of events depicting a potential attack or a malicious activity:-

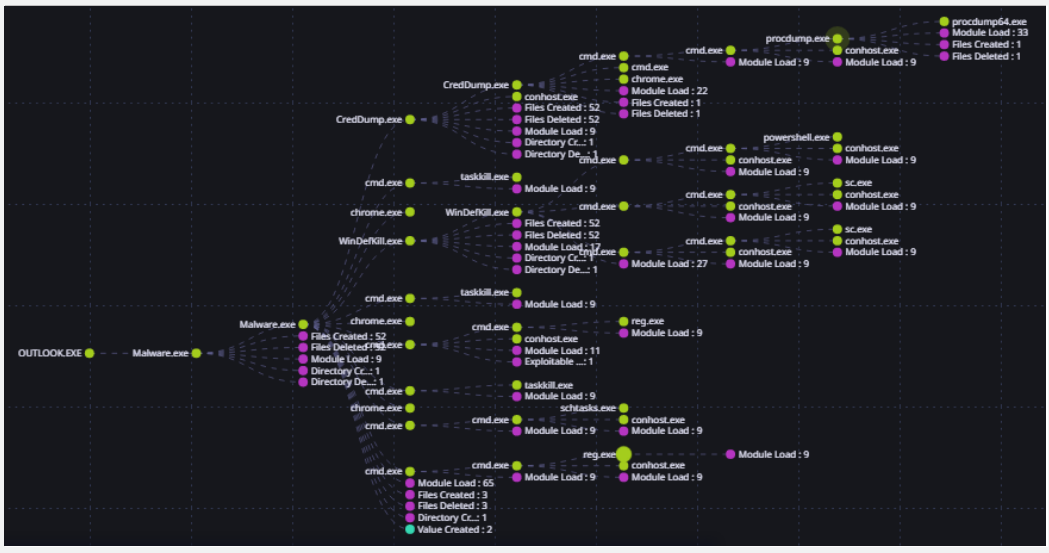

Fig. Sample of HawkkHunt (Seqrite's EDR solution) illustrating process-flow

Advanced EDR solution to the rescue

01 Initial Access

A user might receive an email with some attachment to it or a link in it. This is an attempt by an attacker to try to get into the user's machine. Just sending an email isn't enough. The adversary may rely upon a user opening a malicious attachment to gain execution for the attack to be successful.

Most commonly, the mailbox is accessed through widely used browsers like Google Chrome, Firefox, Microsoft Edge, etc., and particular mailing applications like Outlook, Gmail, etc.

▶ EDR Alert at Initial Access:

Organizations can use Seqrite HawkkHunt's message analysis techniques to check the authenticity of an email. Reputation analysis of the sender and URL analysis mechanisms help raise an alert at the Initial Access level itself. Early-stage detection of such malicious activity effectively circumvents a potentially more significant attack.

02 Execution

Execution can happen in multiple ways. The users' conscious performance accounts for a significant proportion of executions. When a user clicks on the attachment/link, specific processes initiate in the background. These processes perform further attacks without the user knowing about them.

▶ EDR Alert at Execution:

HawkkHunt keeps a close eye on all the processes and their command lines. Usage of scripts to execute the malware or download the malware. HawkkHunt keeps a tab on these processes and raises an alert when such process sequences are observed.

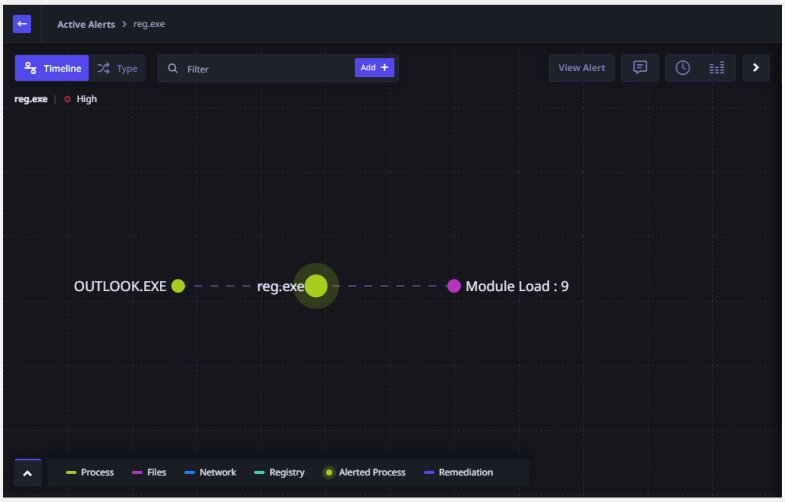

Fig. Process flow for execution.

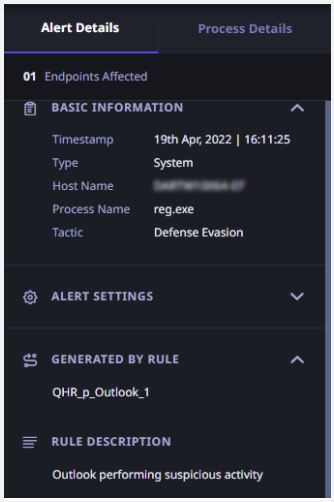

Fig. Alert on Execution

03

Persistence

Malware authors use multiple ways to make the malware persist on the system. Scheduling a task is one of the common ways of achieving persistence. Here, the adversary prepares the execution of the malicious file. Hence, no user interaction is needed to execute the malicious file. For achieving this, the operating system's utilities, such as schtasks.exe, can be used to add the entries in Task Scheduler.

EDR Alert at Persistence:

HawkkHunt monitors all the processes and its command lines. Organizations can monitor the usage of schtasks.exe to schedule a task and use the command line to analyze the presence of any remote location or URL, or IP. When a suspicious task appears to be scheduled, HawkkHunt raises an instant alert.

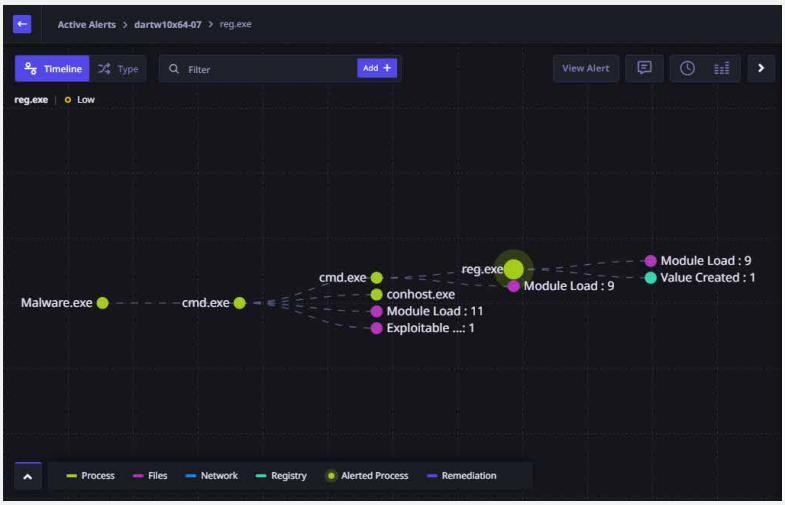

Fig. Process flow for Persistence.

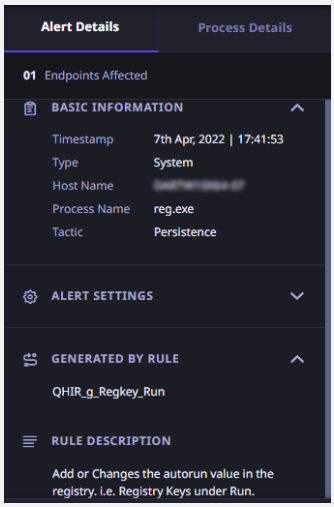

Fig. Alert on Persistence

04 Privilege Escalation

For performing certain activities on the system, the attacker requires high-level permissions. Without these permissions, specific attacks are just not possible. Out of the multiple ways adversaries may use UAC bypass mechanisms is to elevate process privileges on the system. Windows User Account Control (UAC) allows a program to elevate its privileges (to perform a task under administrator-level permissions), possibly prompting the user for confirmation. An unsuspecting user's acceptance of the prompt to escalate privileges results in the malware gaining capabilities to operate with more rigor and causing more significant harm to the user's system and data.

EDR Alert on Privilege Escalation

Out of the multiple ways an attacker can use to bypass UAC, one is by using eventvwr.exe and registry hijacking. Seqrite HawkkHunt's policies are designed so that any hijacking or modification of essential registry keys raises an alert, thereby enabling the incident responders to take quick remediation action.

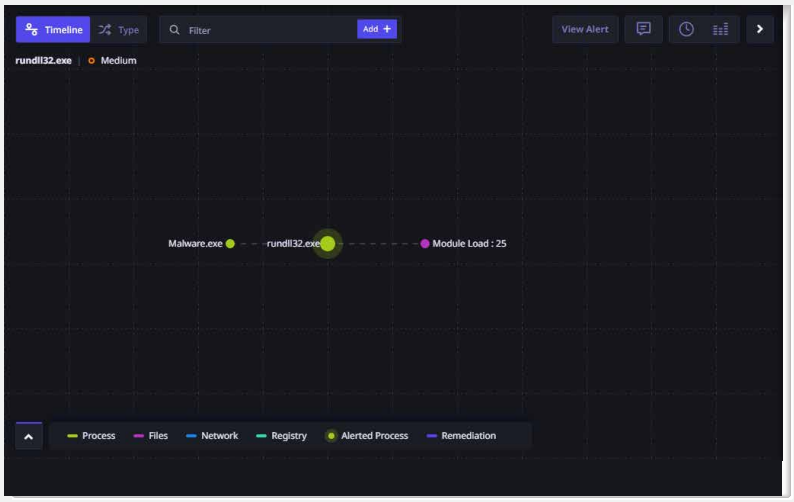

Fig. Process flow for Privilege Escalation.

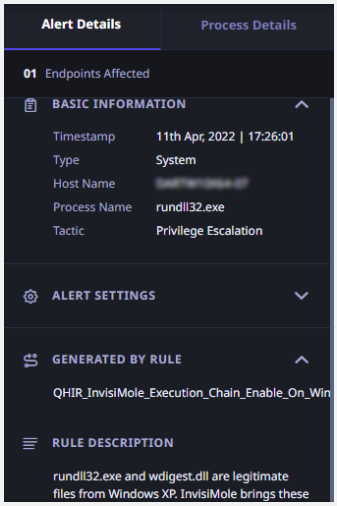

Fig. Alert on Privilege Escalation

05 Defense Evasion

The attackers are very sophisticated nowadays. They have built competencies to determine what all anti-malware services run on the machine. Attackers use various techniques to evade the default defense mechanisms.

EDR Alert on Defense Evasion:

Attackers might employ techniques that stop the Windows Defender service or deletes it altogether to evade detection. HawkkHunt policies are designed in such a way that any changes to the services, such as stopping or restarting a service, get recorded in the events, and an alert is generated if the suspicious event matches the policy.

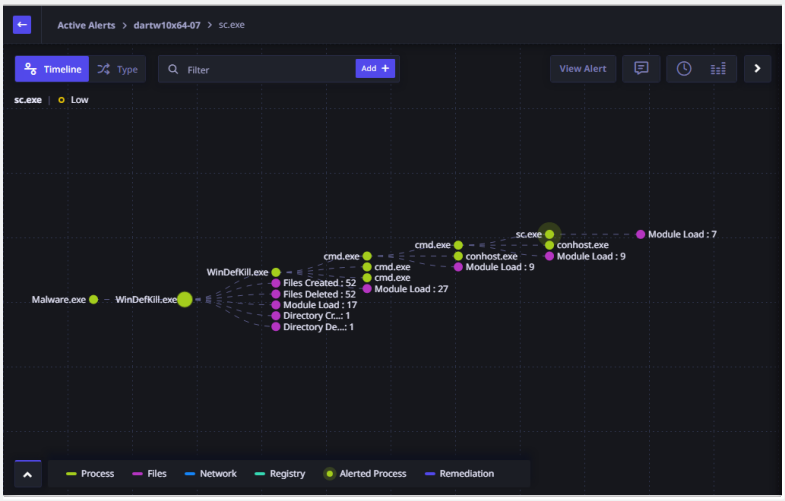

Fig. Process flow for Defense Evasion.

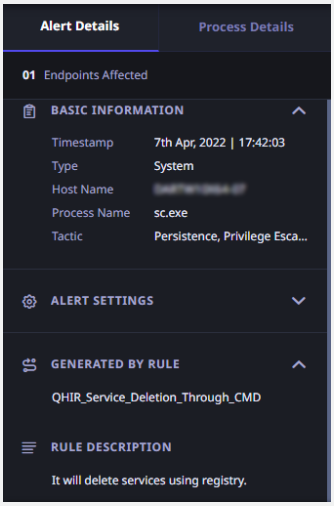

Fig. Alert on Defense Evasion

06

Credential Access

The attacker tries to find the credentials to get hold of the system, i.e., the password to log in to the system. Out of the multiple ways, using LSASS.exe to dump the hash or the password is a commonly used practice. Now, lsass.exe is a simple windows process. So keeping an eye on it becomes necessary because you cannot detect this activity in an Endpoint Security Product.

EDR Alert on Credential Access

Seqrite's HawkkHunt has the capabilities to monitor the process hierarchy. The characteristics of the parent process of lsass.exe play an essential role in alert raising. Keeping an eye on the parent and grandparent of lsass.exe and the command line of lsass.exe can effectively help IRs mitigate the credential access attack. Using these mechanisms, we raise an alert whenever suspicious dumping of credentials is identified.

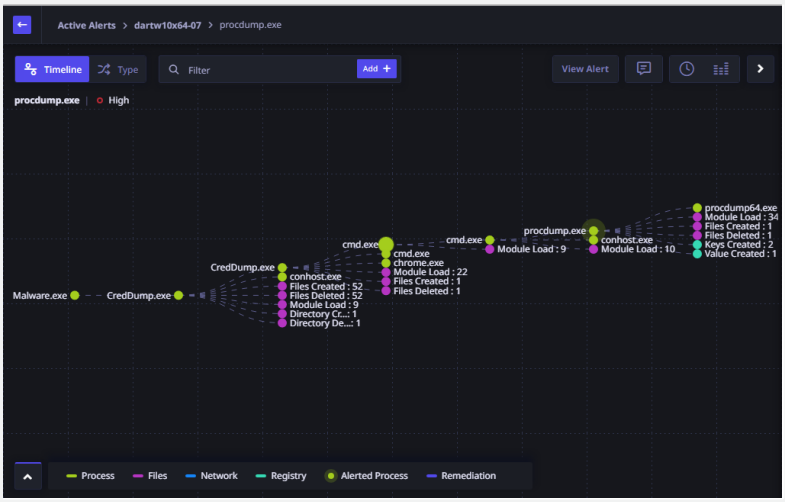

Fig. Process flow for Credential Access.

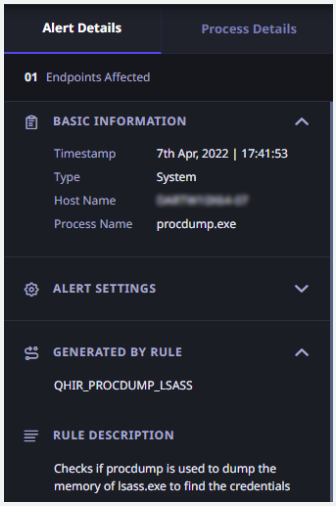

Fig. Alert on Credential Access

07

Discovery

Once the attacker gets control over one system in the network, it tries to get information about other systems in the network. This stage of an attack is termed Discovery. Adversaries may look for details about the network configuration and settings of systems they access or through information discovery of remote systems. Several operating system administration utilities exist that can be used to gather this information.

Examples include ***WMIC, Arp, ipconfig/ifconfig, nbtstat, net, ping, route,*** etc.

EDR Alert on Discovery

HawkkHunt identifies the context of the discovery commands. If the parent or grandparent process of ipconfig, systeminfo, WMIC, or a net command is not out of usual processes, then HawkkHunt raises a red flag.

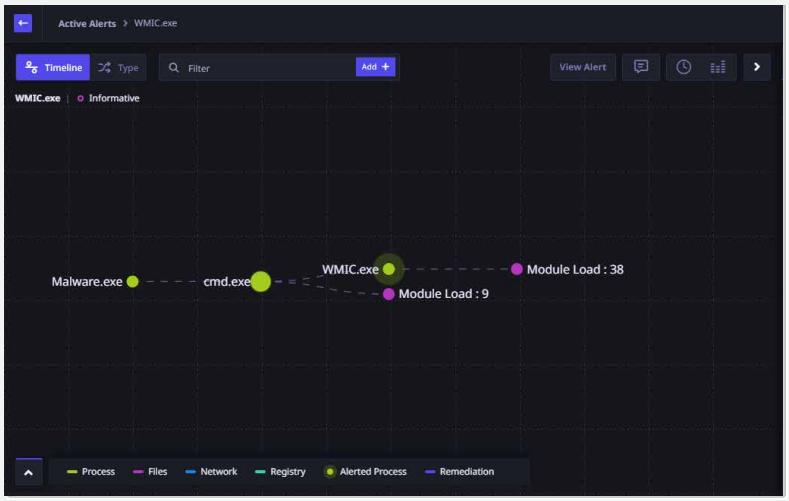

Fig. Process flow for Discovery.

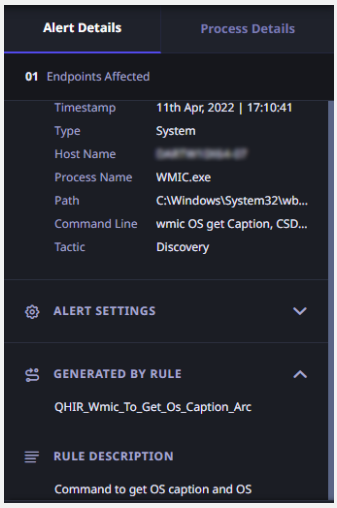

Fig. Alert on Discovery

08

Lateral Movement

Lateral movement is when attackers take control of one asset within your network and then obtain privileged access to move around and exploit other assets.

Once the information regarding its next target in the network is received, the attacker might use tools like PSEXEC to copy and execute the malware on the next target machine.

EDR Alert on Lateral Movement

Seqrite HawkkHunt tracks various tools and their command line parameters for lateral movement. Also, we use super rules, which throw an alert by correlating events from the source and destination machine.

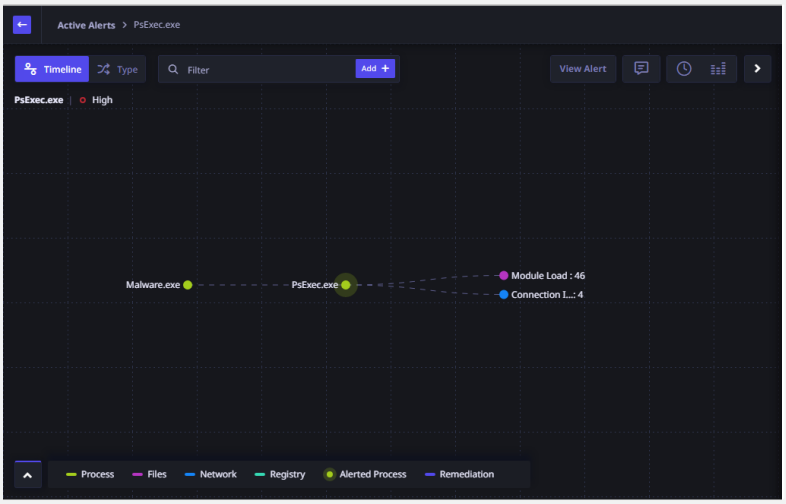

Fig. Process flow for Lateral Movement.

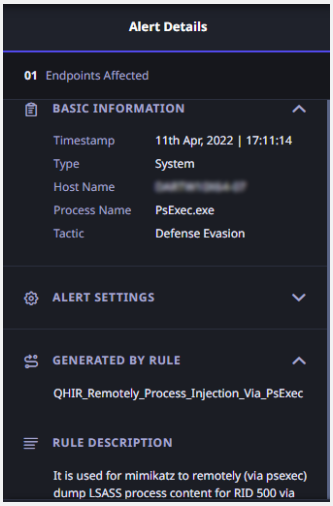

Fig. Alert on Lateral Movement

Conclusion

The traditional reactive approach won't be an effective defense mechanism to tackle the growing risk of cyber-attacks. An effective security posture should shift towards a proactive approach where pre-emptive measures are taken to safeguard our network, systems, and data. Along with the detection mechanisms, prediction mechanisms and policies need to be used to contain the overall risk of sophisticated and targeted attacks. Seqrite HawkkHunt provides this vision and mechanism. As shown in the illustrative figures and screenshots above, Seqrite HawkkHunt provides the incident responders and Admins a much-needed avenue to view a transparent chain of processes, enabling them to conceive and deliver quick and effective remediation measures.

Subject Matter Expert:

Akshay Gaikwad, Nihar Deshpande

SEQRITE

Marvel Edge, Office No. 7010 C & D,
7th Floor, Viman Nagar, Pune - 411014, India.

www.seqrite.com