Attack Graph Emulating the Conti Ransomware Team's Behaviors

By AttackIQ Adversary Research Team

Published: 2022-06-16 · Archived: 2026-04-05 23:51:26 UTC

Sectors targeted: *healthcare; indiscriminate*

Modern cybercriminal syndicates are complex organizations. They are often comprised of merged groups, some have subsidiary organizations within them, they perform a diverse set of operations, and, in the case of Russia-based cybercriminals, often maintain a shadowy affiliation with the Russian government's intelligence services. Since April 2017, the Russian-aligned Conti ransomware-as-a-service (RaaS) operation has been one of the most aggressive and successful ransomware operations, compromising and extorting over 1,000 victims with payouts exceeding $150-180 million USD according to the FBI as of January 2022, leading them to deem it the costliest ransomware strain ever documented. Other sources, however, put their take closer to $2.7B USD, depending on crypto trading price volatility at various times and number of known wallets counted.

On May 19, 2022, the Conti operation officially disbanded, taking down key infrastructure and informing their team leaders that the brand no longer existed. There were indications of a possible breakup beginning late 2021 after Conti's acquisition of the TrickBot malware and operation team, the plans to swap TrickBot with the stealthier BazarBackdoor malware, and then the ransoming of the San Francisco 49'ers, publicly confirmed hours before the U.S. NFL Superbowl on February 13, 2022, but using the smaller *BlackByte* ransomware group as a shell to publicly process the breach on their behalf. Some of their former members have already migrated to these smaller ransomware groups like Karakurt and BlackByte.

During its run, the Conti operation experienced a total of three revenge-based leaks of various combinations of source code, chat logs and technical manuals from insiders, with the last leak entangled in geopolitics surrounding Russia's invasion of Ukraine. The second leak of source code, tools, post-compromise technical manuals and training manuals on August 5, 2021, is the one on which AttackIQ's new attack graph is based due to the fidelity of information available from that leak, particularly in the post-compromise technical manuals used by the ransomware operators. This attack graph emulates the actor's full attack life cycle to help customers validate their security posture against similar attacks.

Despite its break-up, Conti's successful post-compromise tactics, techniques, and procedures (TTPs) employed by the group's operators will live on as these criminal hackers splinter and join new groups, taking their same skills with them and making it vital that organizations continue to be prepared to deal with intrusions using these playbooks despite inevitable minor variations. Validating your security program performance against this type of attack is crucial in reducing risk by increasing resilience. By using this new attack graph in the AttackIQ Security Optimization Platform, security teams will be able to:

1. Evaluate security control performance against attacks using techniques that have had significant real-world impact.
2. Assess security posture against the likely tactics used by Conti's splinter cell groups.

3. Continuously validate detection and prevention pipelines against activity that leads to destructive actions.

Attack Graph Emulation of Conti Techniques

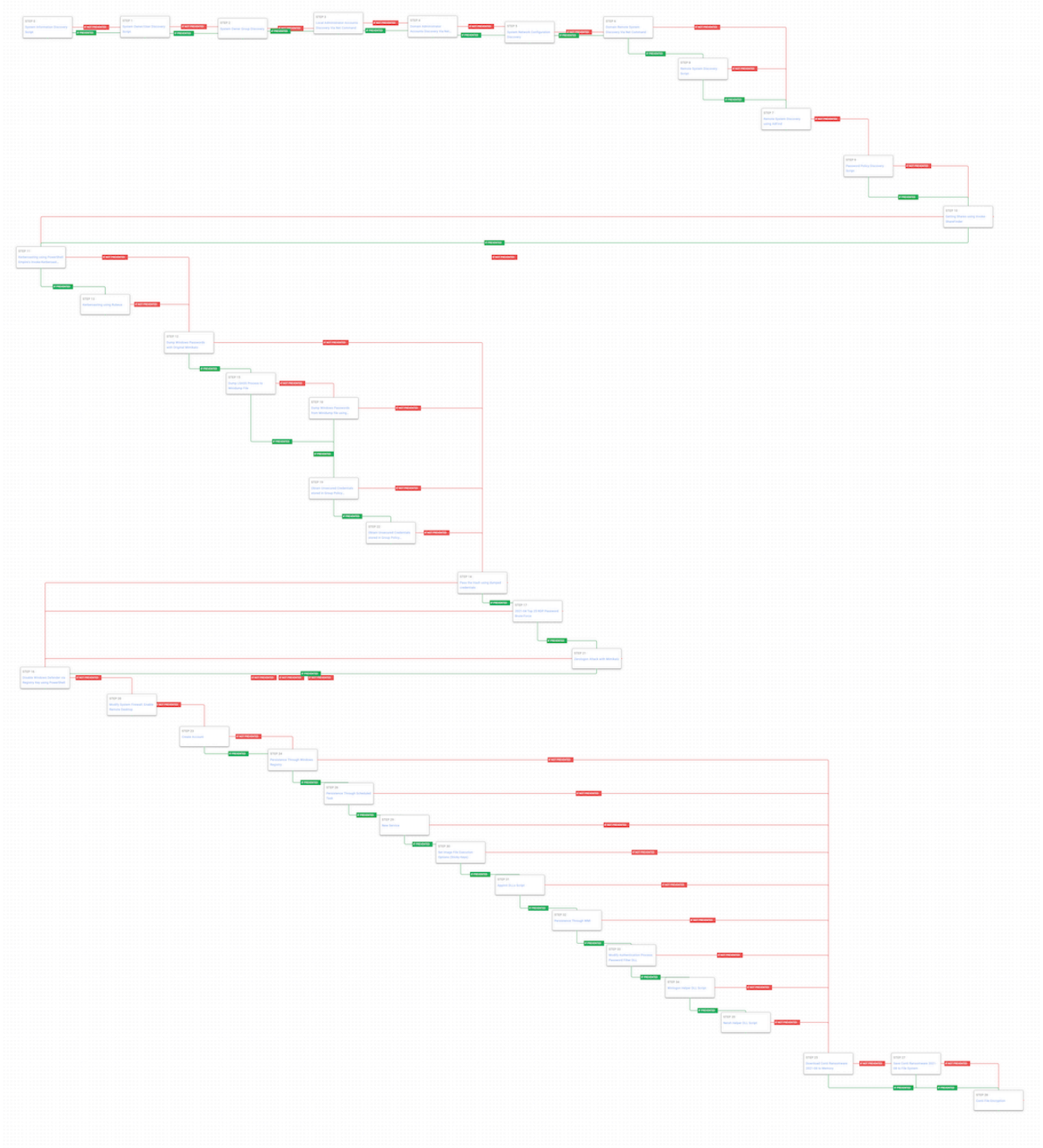

Figure 1 – Conti Attack Graph

Conti ransomware operators would gain an initial foothold to victims from various initial access brokers (IABs) such as FIN12. Typically, this would start from spear-phishing campaigns that delivered TrickBot and IcedID. Our attack graph begins after the initial access has been achieved and the Conti actors are about to take hands-on-keyboard control of the compromise.

The actors' first steps are to explore and learn more information about the initial host that has been compromised. Conti runs a series of discovery techniques by using native system commands to live off the land and try to not draw additional attention.

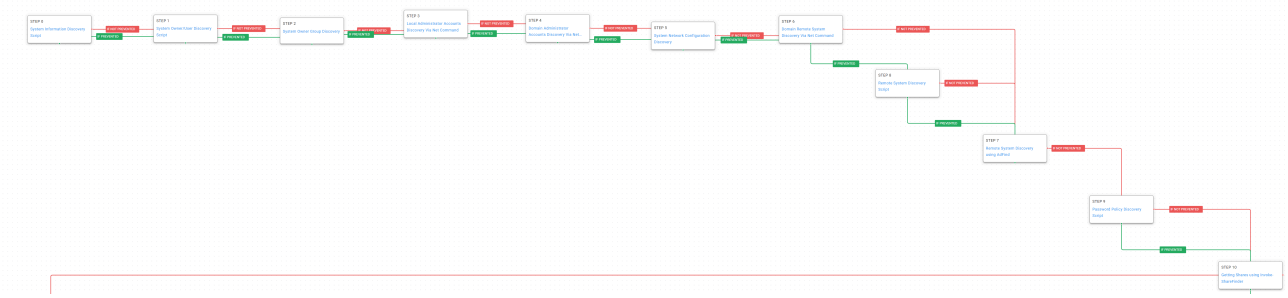

Figure 2 – Discovery

System Information Discovery (T1082): Execute native commands like "systeminfo" or "lshw" to learn about the system confirmations.

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

Account Discovery – Local Account (T1087.001): On Microsoft Windows hosts use the "net localgroup administrators" command to identify any local administrator accounts.

System Network Configuration Discovery (T1016): Run built-in tools and commands to get routing, network adapter, network shares, and connected domain controller information.

Remote System Discovery (T1018): Search for other domain computers using the "net group" command. If the activity is prevented, downloading and leveraging the Adfind utility is attempted. Due to licensing and distribution issues, the "Remote System Discovery Using AdFind" Scenario will need to be manually configured with a locally uploaded "AdFind" binary. This binary can be found and downloaded here at the "Download" portion of the page.

Password Policy Discovery (T1201): Conti will want to understand the local and domain password requirements to help validate potential stolen or harvested credentials.

Network Share Discovery (T1135): The threat actors leverage the Invoke-ShareFinder cmdlet from Veil-PowerView to identify remote network shares that could contain files of interest.

After completing the discovery phase of their attack, Conti will move on to gaining access to more credentials. They use a variety of techniques to gain not only other local accounts but remote and service accounts that will allow them to move laterally.

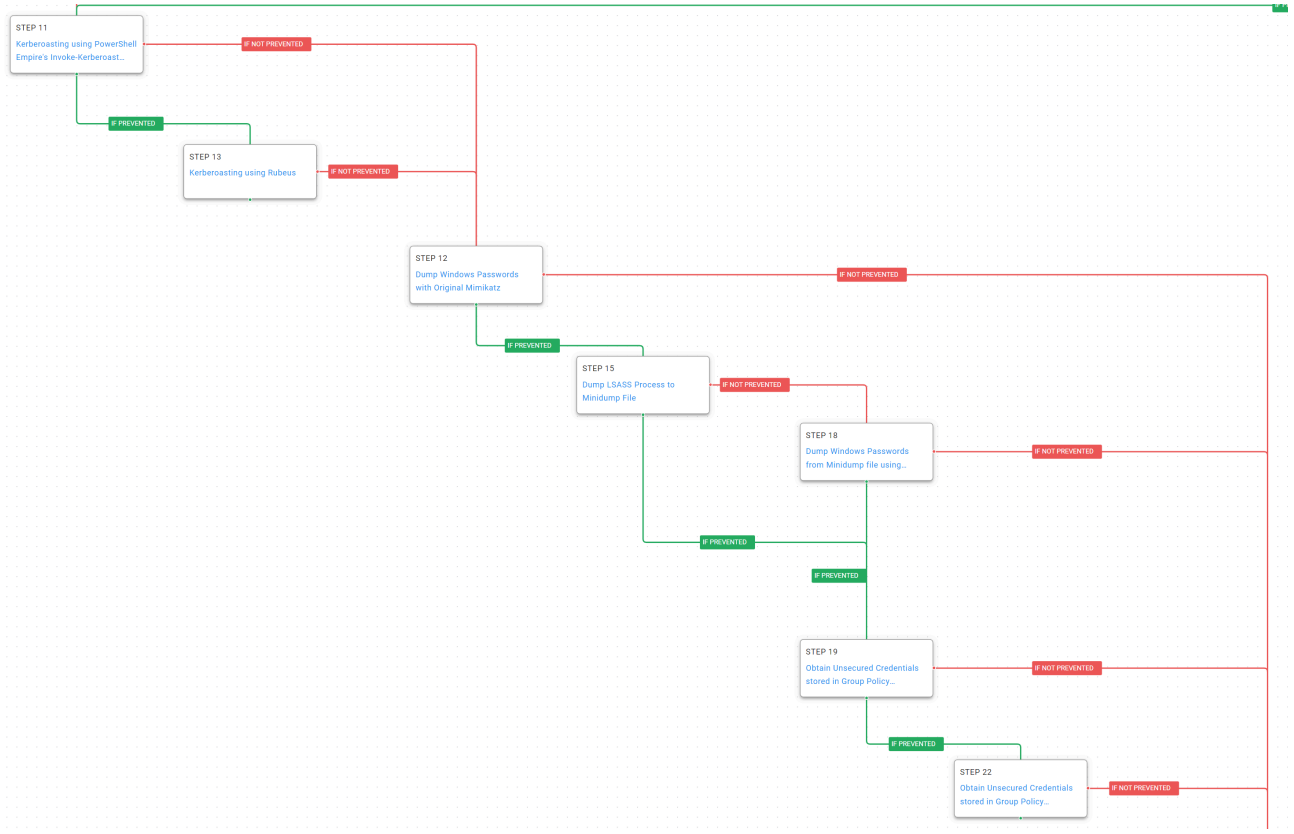

Figure 3 – Credential Access

Steal or Forge Kerberos Tickets – Kerberoasting (T1558.003): Kerberoasting allows an attacker to attempt to extract password hashes for accounts using their Service Principal Name (SPN) ticket. The attack graph first tries using the Invoke-Kerberoast PowerShell cmdlet and if prevented tries using Rubeus.

OS Credential Dumping (T1003): Conti leveraged Mimikatz to dump credentials on Windows hosts.

OS Credential Dumping – LSASS Memory (T1003.001): The Local Security Authority Subsystem Service (LSASS) has credentials in its memory. The process address space is dumped to a minidump and then Mimikatz is used to extract credentials.

OS Credential Dumping – DCSync (T1003.006): Mimikatz is able to impersonate a Domain Controller and request password with domain replication.

Unsecured Credentials – Group Policy Preference (T1552.006): Group Policy Preference XML files contain encrypted credentials for creating local accounts or mapping network drives. These files can be collected from domain connected computers and the credentials harvested for the local accounts.

Armed with additional credentials, the Conti ransomware operators will begin moving laterally to identify and find additional targets of interest.

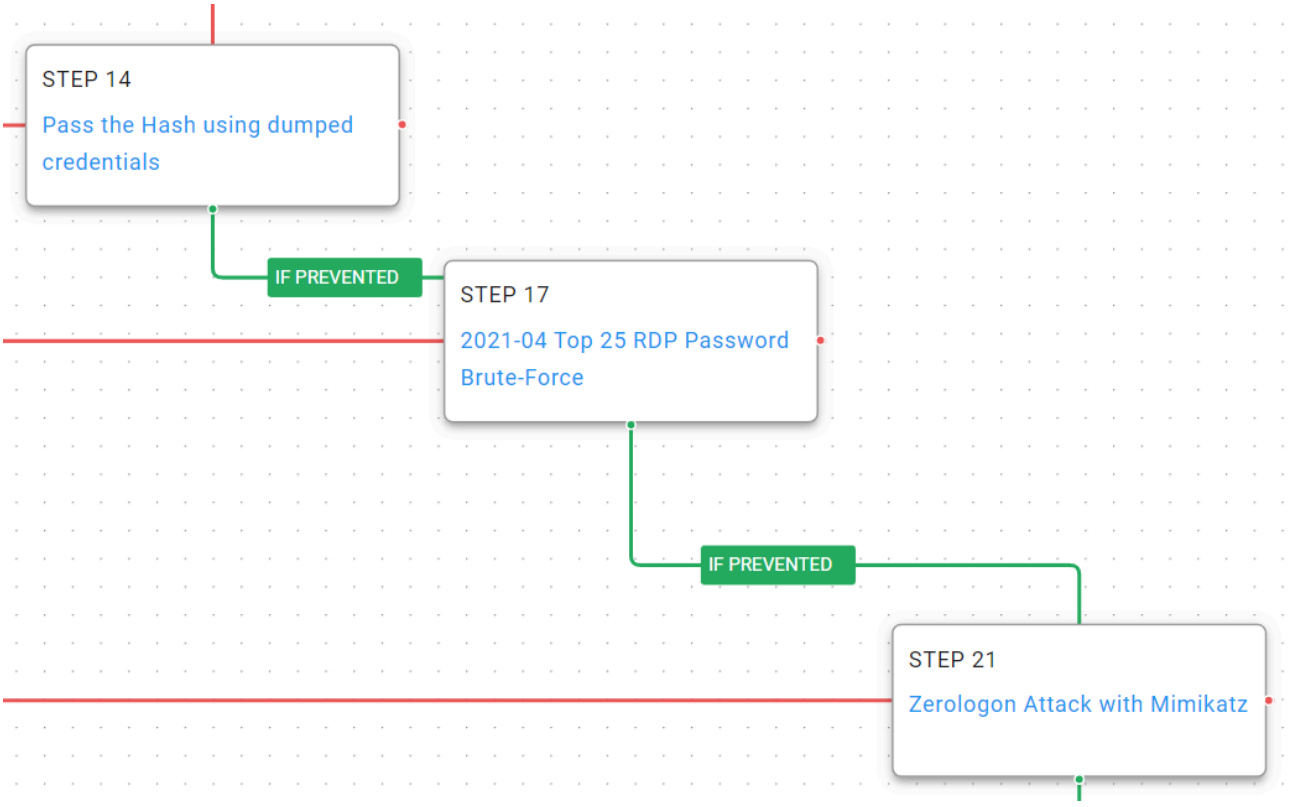

Figure 4 – Lateral Movement

Pass the Hash (T1550.002): Using the hashed credentials obtained in the previous attack phase, they can be used to authentication via NTLM to other enterprise resources.

Exploitation of Remote Services (T1210): Using ZeroLogon Conti can easily impersonate any domain computer and take control of the domain admin account when used against vulnerable domain controllers.

Brute Force (T1110): When all else fails, attempt to brute login using RDP to remote systems with a username and password dictionary.

After gaining access to more of the victim's network the actor is going to want to disable or circumvent security controls to make future actions easier and harder to detect and prevent.

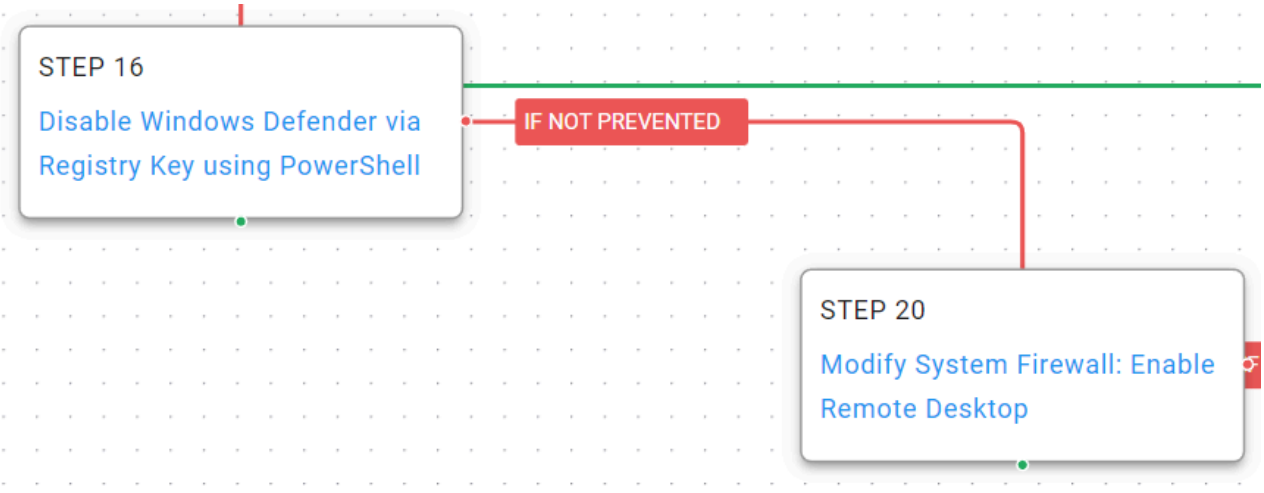

Figure 5 – Defense Evasion

Disable or Modify Tools (T1562.001): The actor uses PowerShell to disable Windows Defender.

Disable or Modify System Firewall (T1562.004): The local host firewall is configured to allow for incoming RDP connections so the actors can move away from interacting with their backdoor and begin using the native Windows functionality.

Conti will begin establishing alternative persistence mechanisms across many of the compromised hosts to decrease the likelihood that the victim organization will be able to quickly remediate their access. The actor uses a variety of methods and tries alternatives until successful. These methods are used to maintain access to their backdoors or ensure that the encryption process continues to run even after interruption.

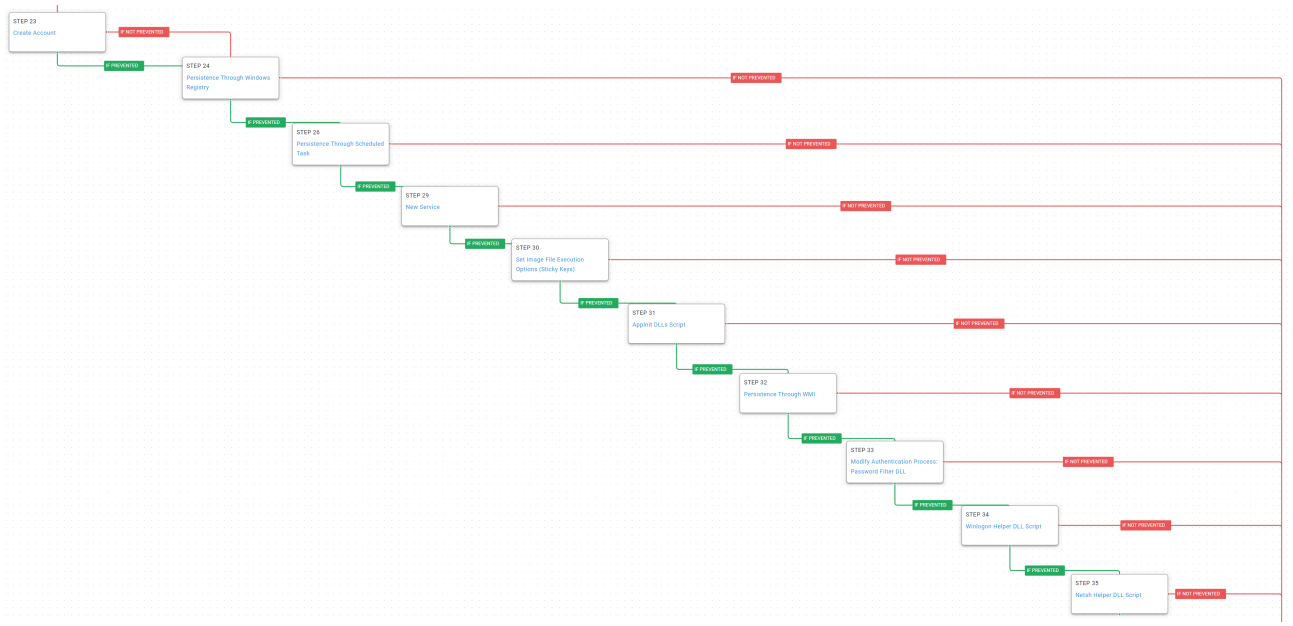

Figure 6 – Persistence

Create Local Account (T1136): Conti will create a new account to side-step any enterprise password resets for known legitimate accounts.

Registry Run Keys (T1547.001): Windows has many registry keys that identify applications or commands to be run at startup.

Scheduled Task / Job – At (T1053.002): The “at” command is used to create a scheduled that can re-launch Conti’s malware periodically or after a restart.

Windows Service (T1543.003): Conti creates new Windows services to re-launch their backdoors after a restart.

Event Triggered Execution – Accessibility Features (T1546.008): Set the debugger for the Sticky Keys helper application to launch a command shell when a user accidentally hits the shift key too many times in a row or if the accessibility feature has been enabled.

Event Triggered Execution – AppInit DLLs (T1546.010): By creating an AppInitDLL entry in the registry, Conti ensures their malicious DLL file is loaded into every process as boot.

Windows Management Instrumentation (T1047): WMI can be used to launch an executable or command when a common event consumer is trigger.

Modify Authentication Process – Password Filter DLL (T1556.002): Microsoft allows for password filters to be configured for domain and local accounts that enable stricter password policies. By installing their own password filter Conti is able to receive the plain text passwords for accounts from the Local Security Authority (LSA).

Winlogon Helper DLL (T1547.004): Conti configures registry keys that force malicious DLL files to be loaded with the Winlogon.exe process.

Netsh Helper DLL (T1546.007): Netshell (netsh) is a command line tool that facilitates interactions with a system's network configurations. It can be configured via registry entries to load malicious DLL files every time "netsh.exe" is executed.

At this point Conti has everything they need to complete their goal and begin encrypting the hosts.

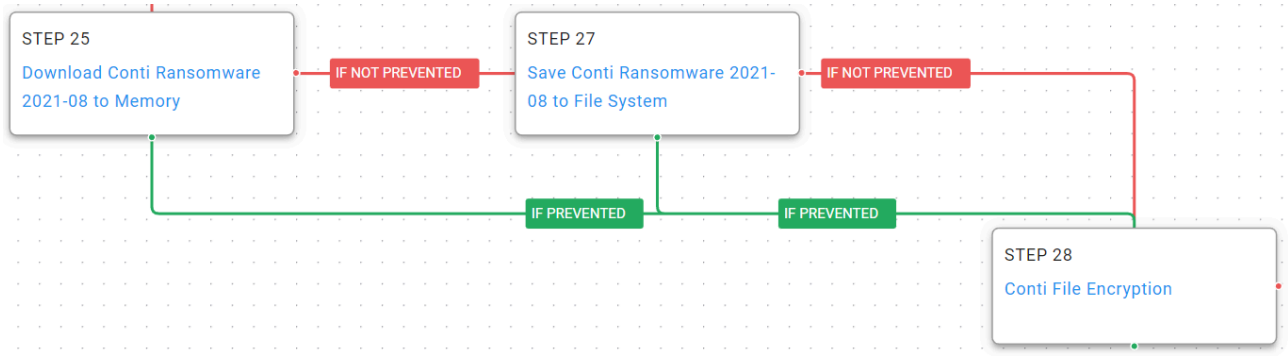

Figure 7 – Encryption

Ingress Tool Transfer (T1105): Download and save samples of the actor's Conti ransomware malware to disk.

Data Encrypted for Impact (T1486): AttackIQ has replicated the functionality used by the Conti ransomware to encrypt files on the targeted hosts. This includes the common file extensions and encryption methods utilized by the actor.

Detection and Mitigation Opportunities

With so many different techniques being utilized by threat actors, it can be difficult to know which to prioritize for prevention and detection opportunities. AttackIQ recommends first focusing on the following techniques emulated in the Conti attack graph before moving on to the remaining techniques.

1. Ransomware Encryption. It should go without saying that as a last resort, preventing your systems and files from being encrypted should be your number one focus. Ensuring that you have the layered endpoint defenses including Antivirus and EDR solutions is critical.

1A. Detection Process

Ransomware attacks are best prevented and alerted by your EDR/AV Policies. Typically, a configuration for ransomware protection is presented and we strongly encourage that it is enabled in your security controls. There are three telling signs of ransomware activity in an environment that you could query for and possibly make preventative detections if your security controls allow. Those three are **deletion of shadow volumes**, **suspicious amounts of exfiltrated data**, and of course, wide set **file encryption**.

Detecting deletion of shadow volumes:

Via vssadmin.exe:

```
Process Name == (cmd.exe OR powershell.exe)
Command Line CONTAINS ("vssadmin" AND "Delete Shadows")
```

Via vssadmin.exe:

```
Process Name == powershell.exe
Command Line == "Get-WmiObject Win32_Shadowcopy | ForEach-Object ($_.Delete();)"
```

Detecting suspicious Data Exfiltration:

Detecting exfiltration is well suited for IDS/IPS and DLP solutions. These products should be configured to identify sensitive files. If sensitive files, or a large amount of web traffic is sent to a rare external IP, it should be detected or prevented depending on security policies for the security control. Historical NetFlow data logging can also bubble up hosts that are experience uncommon peaks in outgoing traffic.

Detecting Ransomware-like File Encryption:

Utilizing an EDR or SIEM/SOAR product can help detect and prevent suspicious file encryption related to ransomware attacks. Utilizing these tools to look for excessive file modifications (greater than 1000 on a system) within less than a minute of time is a good starting indicator. To increase the fidelity a bit, you could include file modification file extension to popular ransomware extensions such as .conti, .Locky, .Ryuk, etc. If possible, with a SOAR or preventative EDR platform, we recommend setting these detections to kill all processes involved in creating the alert as it will most likely stop the spread of the Ransomware.

1b. Mitigation Policies

MITRE recommends the following mitigations for Data Encrypted for Impact (T1486):

- M1040 – Behavior Protection on Endpoint
- M1053 – Data Backup

2. Common Persistence Mechanisms

Similar to our recommendation in the Karakurt Data Extortion blog, being able to stop an actor from creating additional hooks into your environment is critical in ensuring that their access can be fully removed during your initial remediation processes.

Focusing on the most common persistence mechanism like Windows Service (T1543.003), Registry Run Keys (T1547.001), and Scheduled Tasks (T1053.002) should be your first targets for prevention and detection.

2a. Detection Process

Detecting Windows Services Persistence:

```
Process Name == (cmd.exe or powershell.exe)
Command Line CONTAINS ((‘sc’ or ‘sc.exe’) AND ‘create’ AND ‘binpath=”<path to trusted executable>”’
AND start=”auto”)
```

Detecting Registry Run Keys Persistence:

```
Process Name == (cmd.exe or powershell.exe)
User NOT IN <list of expected reg.exe users>
```

Command Line CONTAINS((reg or reg.exe) AND (“HKEY_CURRENT_USER” OR “KEY_CURRENT_MACHINE”) AND “\SOFTWARE\Microsoft\Windows\CurrentVersion\” AND (“run” OR “runonce”))

Detecting Scheduled Tasks Persistence:

```
Process Name == (cmd.exe OR powershell.exe)
Command Line CONTAINS (“schtasks” AND “/create” AND (“cmd” OR powershell”) AND (“.exe” OR “.bat”) AND
“/ru system”)
```

2b. Mitigation Policies

MITRE recommends the following mitigations for Create or Modify System Process: Windows Service (T1543.003):

- M1040 – Behavior Protection on Endpoint
- M1026 – Privileged Account Management
- M1022 – Restrict File and Directory Permissions
- M1018 – User Account Management
- M1047 – Audit

Recommendations for mitigation of Boot or Logon Auto-start Execution: Registry Run Keys/Startup Folder (T1547.001):

It is recommended that only administrators and end users with a specific need be able to run administrative tools such as cmd.exe, powershell.exe, reg.exe, and regedit.exe. Limiting these tools to only authorized users greatly reduce the chance of a compromised end user being able to modify the registry for persistence.

MITRE recommends the following mitigations for Scheduled Task/job: At (T1053.002):

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

Multiple Native Discovery Commands

One of the very first things that most actors will do when first gaining access to a compromised host is use the native tools and commands available in the operating system to learn more about the host and its connected network. By using built-in functional the actor will blend in with legitimate user traffic. The trick is to look for the series of different discovery commands being executed in short time windows.

Multiple commands in time windows can be an initial suspicious indicator that an actor may have an established a foot hold and can be used to increase the fidelity of future additional suspicious activities that are observed from a single host.

3a. Detection Process

With an EDR or SIEM product, you could easily detect when administrative commands are run by non-authorized / expected users:

```
Process name == (“cmd.exe” OR “powershell.exe”)
User NOT IN (<list of expected administrators and power users>)
```

Command Line CONTAINS (“systeminfo” OR “whoami” OR “net users” or “net localgroup Administrators” OR “route print” OR “ipconfig /all” OR “arp -a” OR “wmic ntdomain” OR “wmic netuse” OR “wmic nicconfig” OR “Get-ADComputer” OR “net accounts” OR “Invoke-ShareFinder”)

3b. Mitigation Policies

It is recommended that only administrators and end users with a specific need be able to run administrative tools such as cmd, powershell, net, and wmic, systeminfo, arp, or route. Limiting these tools to only authorized users greatly reduces the chance of a compromised end user being able to enumerate system and environmental settings.

Conclusion

In summary, this attack graph will evaluate security and incident response processes and support the improvement of your security control posture against an actor with focused operations to encrypt your systems and data. With data generated from continuous testing and use of this attack graph, 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.

Source: https://attackiq.com/2022/06/15/attack-graph-emulating-the-conti-ransomware-teams-behaviors/