

Active Defense, Offensive Countermeasures, and Cyber Deception

John Strand | Bryce Galbraith | Paul Asadoorian

- *Course Virtual Machines*



Course Virtual Machines

- Let's get to know the VM



This Course Is Different

- This course is different from other courses...
 - The concepts, the approach, the labs
 - Most of the labs are *not in the slides* (because we like you :-))
 - This makes them more accessible *after* class, when you need them most
 - All labs using the VM are inside the VM, within github
 - This means you do not have to dig through hundreds of pages to figure out how something works later
 - There are also prerecorded video walkthroughs of each lab on the USB and embedded in Discord!
- You're welcome! Enjoy! ;-)



MITRE Engage

Filter by ATT&CK® Groups

Group

Filter by ATT&CK® Tactics

Tactic

Filter by ATT&CK® Techniques

Technique

Expose	
Collect	Detect
API Monitoring	Introduced Vulnerabilities
Network Monitoring	Lures
Software Manipulation	Malware Detonation
System Activity Monitoring	Network Analysis

Affect		
Prevent	Direct	Disrupt
Baseline	Attack Vector Migration	Isolation
Hardware Manipulation	Email Manipulation	Lures
Isolation	Introduced Vulnerabilities	Network Manipulation
Network Manipulation	Lures	Software Manipulation
Security Controls	Malware Detonation	
	Network Manipulation	
	Peripheral Management	
	Security Controls	
	Software Manipulation	

Elicit	
Reassure	Motivate
Application Diversity	Application Diversity
Artifact Diversity	Artifact Diversity
Burn-In	Information Manipulation
Email Manipulation	Introduced Vulnerabilities
Information Manipulation	Malware Detonation
Network Diversity	Network Diversity
Peripheral Management	Personas
Pocket Litter	



- *Definitions and Disclaimers*



Disclaimer

- The tactics covered in this course *could* get you into trouble
 - But so can most activities, if not done *properly* (e.g., driving)
- The masses will impulsively state that this is a bad idea...
 - But the masses continue to fail miserably
 - If you want different results, you have to do something differently
- Make sure you vet some tactics with your legal team, human resources, and upper management first
- Get a warrant whenever appropriate
- Maintain high ethical (and legal) standards
- Don't become what you're defending against...



What Is Active Defense?

- Active Defense
 - The employment of *limited offensive action and counterattacks* to deny a contested area or position to the enemy
 - Proactive, anticipatory, and reactionary actions against aggressors
 - The adversaries are already inside your gates...
- Passive Defense
 - Measures taken to reduce the probability of and to minimize the effects of damage caused by hostile action *without the intention of taking the initiative*
 - Traditional static defenses (i.e., hope for the best)
- Prevent | Detection | Respond
 - Prevention is ideal, *but detection is a must*, and detection without response is of little value...

What Are Offensive Countermeasures?

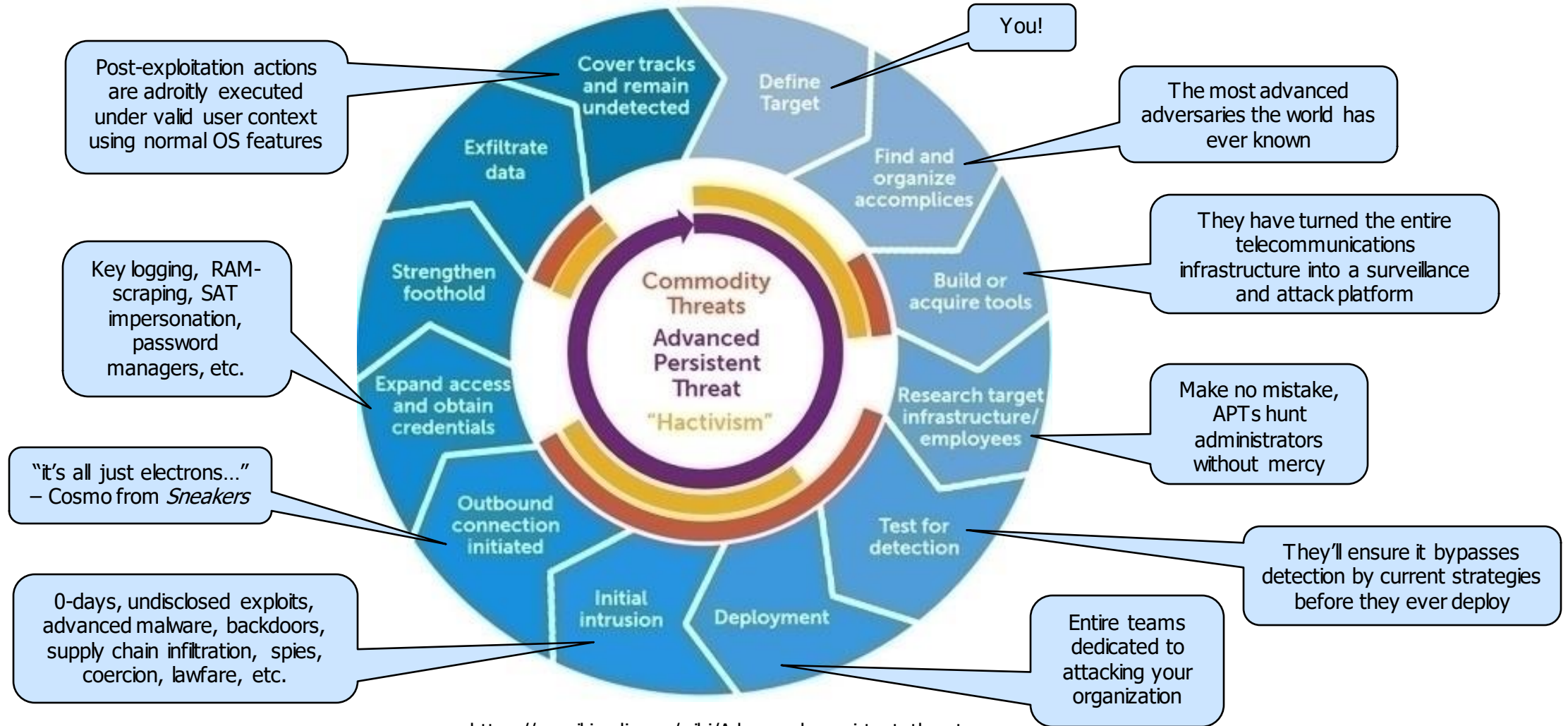
- Offensive countermeasures employ offensive techniques as aggressors attack ... *but with a defensive posture*
 - Aikido provides an excellent analogy
 - Aikido focuses on redirecting and blocking opponents' attacks while taking considerable care not to harm them in the process
 - Aikido practitioners *respond* to attacks; they do not *initiate* attacks
- Think poison, not venom
 - Poison is taken then consumed, whereas venom is injected
 - Lay traps inside *your* systems, but don't attack *theirs*
- Always ensure solid legal footing
 - Proper authorization, warrant, written approval, etc.



What Is Cyber Deception?

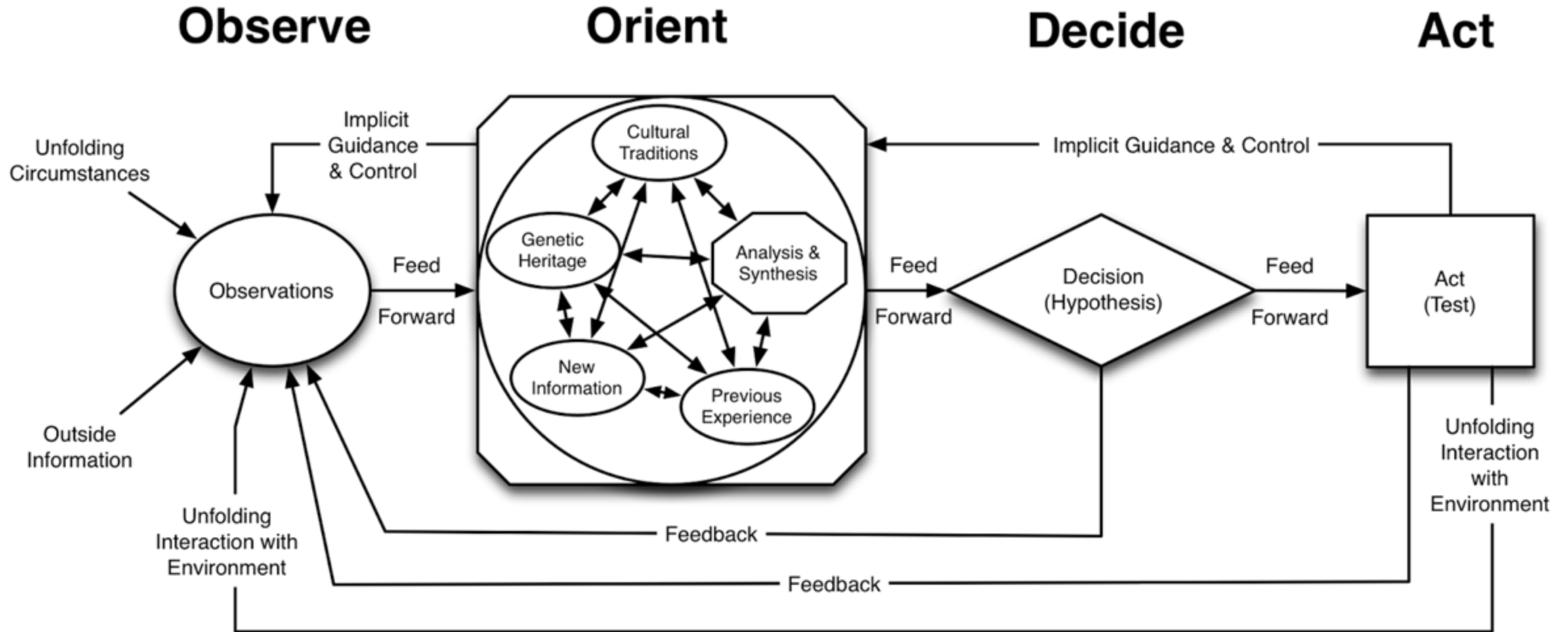
- Cyber deception is the deliberate and calculated process of deceiving attackers in an effort to wage a better defense
 - Slow them down, confuse them, deceive them ... make them work harder
 - Serves to significantly increase your chances of detection
 - Designed to make $\mathbf{Detection}_t + \mathbf{Reaction}_t < \mathbf{Attack}_t$ ($\mathbf{D}_t + \mathbf{R}_t < \mathbf{A}_t$)
- Cyber deception does not replace other efforts or layers of defense
- It should complement and feed the other layers
- Militaries have employed deception strategies since the beginning of time. Why don't we?

“Know Thy Enemy” —Sun Tzu

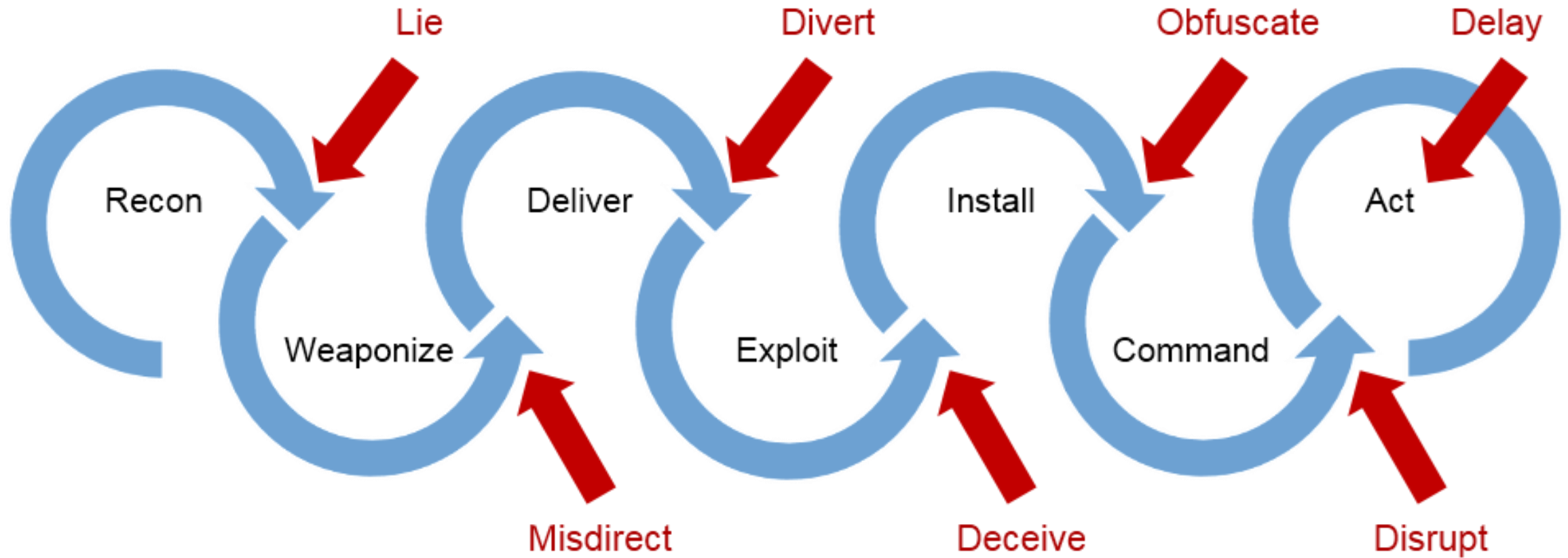


https://en.wikipedia.org/wiki/Advanced_persistent_threat

The OODA Loop



Disrupting the OODA Loop



How to Avoid Legal Trouble

- Not everyone agrees on how to avoid trouble
 - But when does everyone agree on anything?
- A few simple tips go a long way
 - Don't put malware where it is publicly accessible
 - Prevent collateral damage
 - Make the attackers come to you first
- Use warning banners and Terms of Use (TOU)
 - It's not as hard as it might seem at first
 - Cortana is “ready to help you out.” ;-)
- More on this topic later...



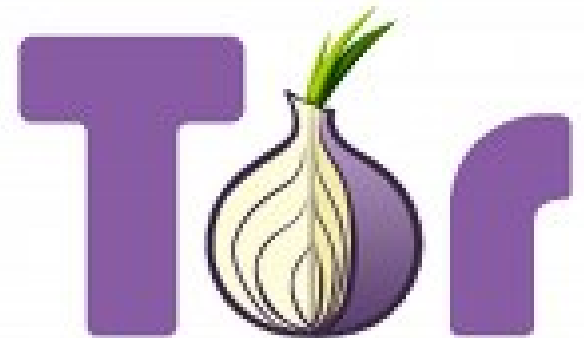
Warning Banners

- It is, however, *illegal* to set up lethal traps for trespassers
 - And this isn't our goal anyway (remember the Aikido analogy)
- You *can*, however, warn them of “evil” things on the network
- Access checks, authentication verification, geo-location, etc.
- Consult with a lawyer and get a warrant



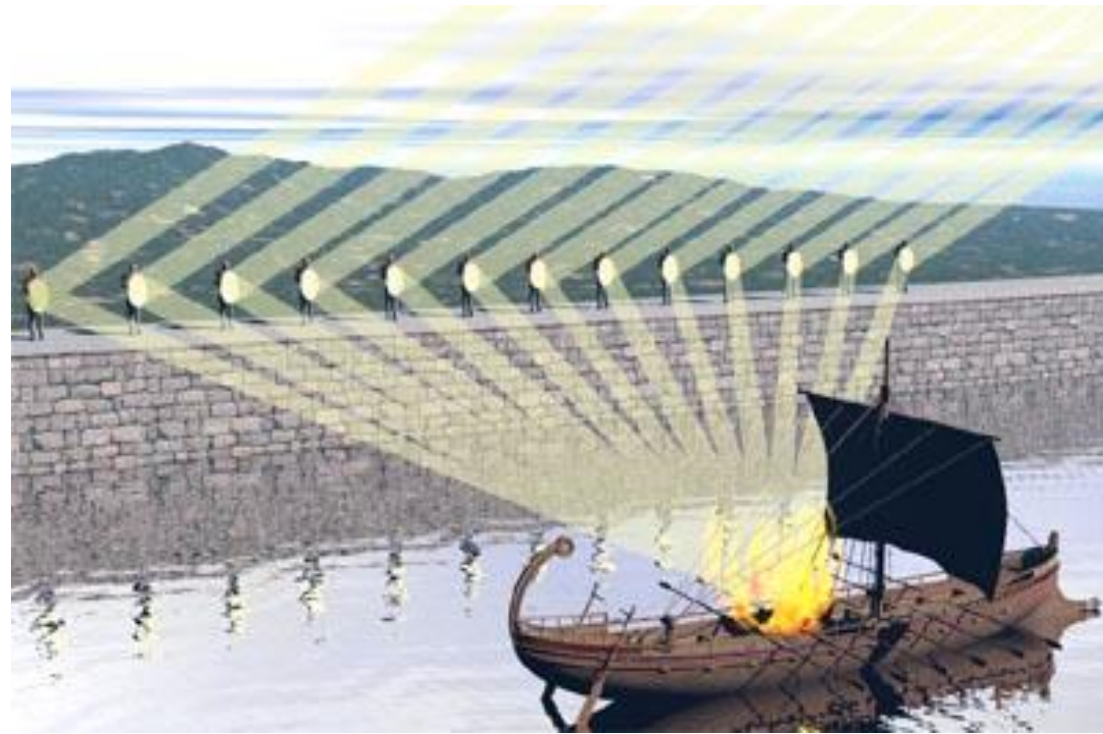
Why These Skills Are Critical

- Eventually, you will need these skills
- Attackers are getting more and more brazen
 - There is very little perceived risk on their part
 - We have rules; they don't
- You might need to figure out what an attacker is seeking
- You might need to gather information about an attacker
 - Attacking from a bot-net
 - Attacking through TOR or I2P

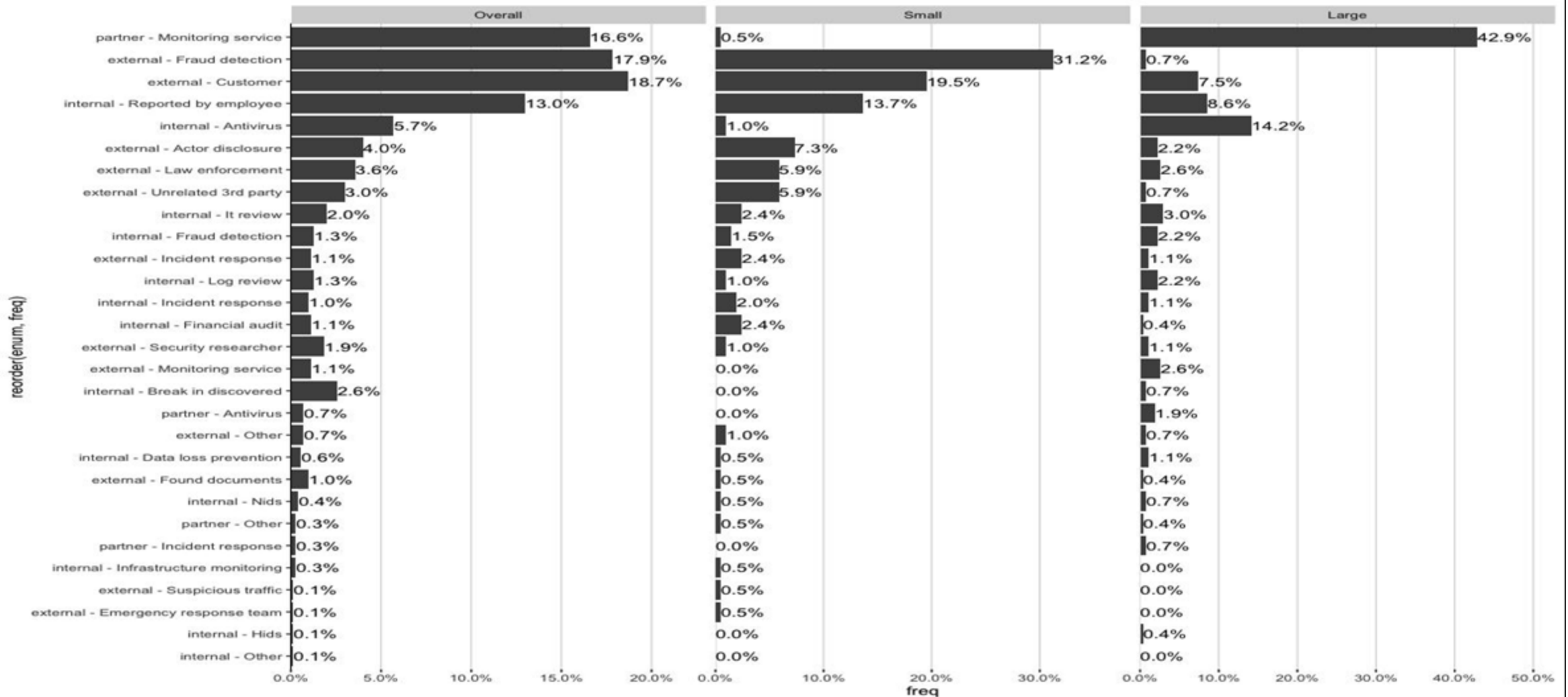


Introductions and Standards

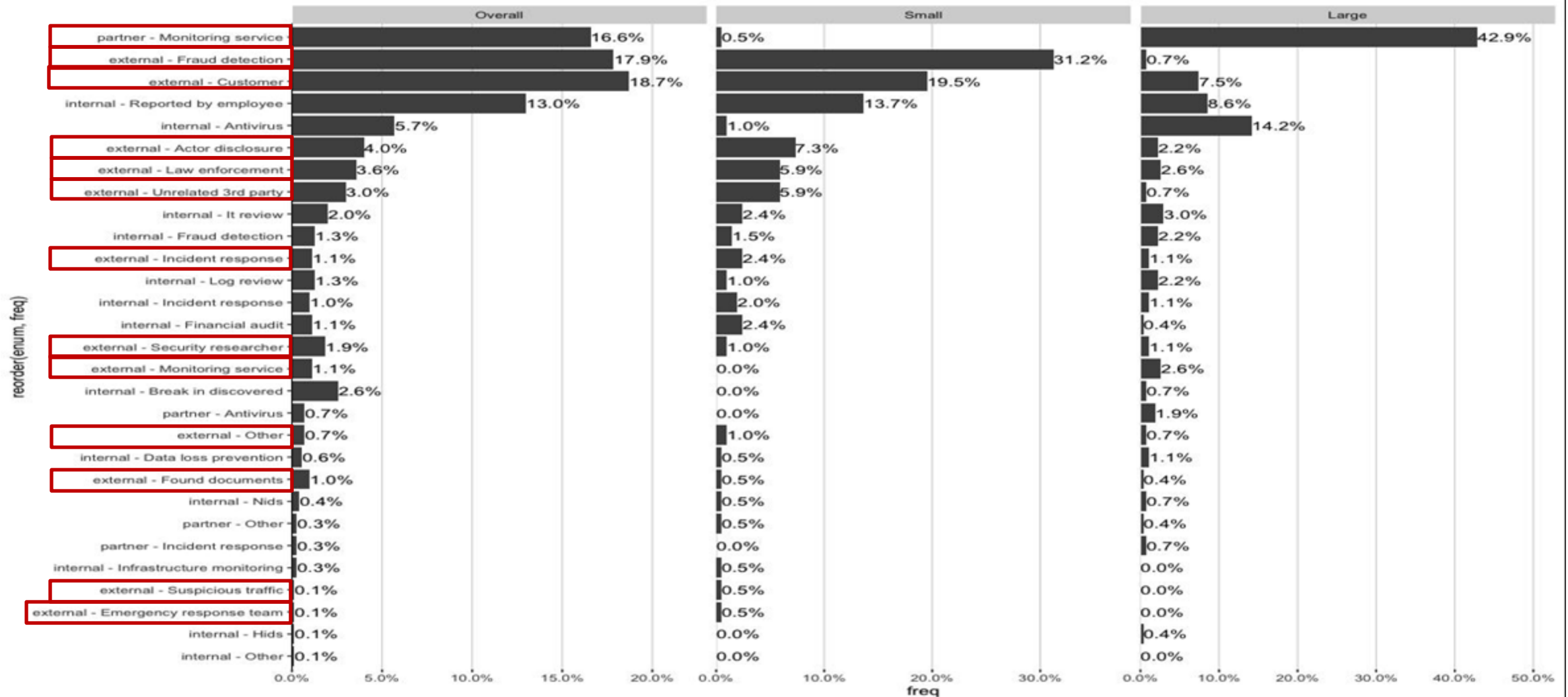
- *Mourning Our Destiny, Leaving Youth and Childhood Behind*



These Are Just The Ones We Know About...



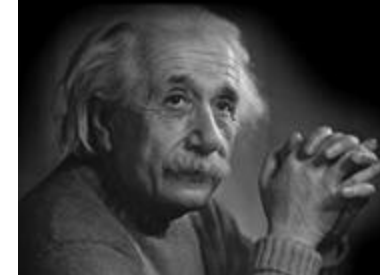
These Are Just The Ones We Know About...



Why Current Strategies Are Not Working

- Go back a few years in your minds...
- What were the recommendations then?
 - Patch, strong passwords, anti-malware, firewalls/proxies, etc.
- What are they saying now?
 - Same things with Next-Gen in front!
 - Next-Gen firewall, Next-Gen anti-malware, and so on...
 - It's gotten better (arguably), but it's reactionary by nature
- Do you see a pattern?

Insanity: doing the same thing over and over again and expecting different results.



Albert Einstein
German Theoretical-Physicist
(1879-1955)

QuoteHD.com

Top Security Product Vendors?

- What are the top three or four AV companies?
- What are the top three or four IDS companies?
- What are the top three or four firewall companies?
- What is their total market share?



Advanced Persistent Thieves (APT's)

- So who's after your electrons?
 - China?
 - Russia?
 - The Five Eyes?
 - Other nation-states?
 - Organized crime?
 - Insiders?
 - **All of the above!?**



Consider Their Capabilities

- Virtually unlimited resources (via taxpayers)
- Direct access to your electrons
- Never-ending exploits/backdoors
- Elaborate anonymization and C2
- Immunity from prosecution
 - Plausible deniability (i.e., lies)
 - Laws are for their subjects, not them...
- Highly motivated/conditioned
 - Feel it is their right/obligation/duty
 - “We do it for [insert reasons here]”



We Should Not Be Surprised

- Most good testing firms are not thwarted by traditional defenses
 - Black Hills Information Security, Layered Security, TrustedSec, and SecureIdeas bypass these defenses as a course of business
- We know nation-states are *at least* as capable (understatement)
- And their budgets eclipse security firms (thanks to taxpayers)
- It's safe to say that nation-states run circles around most defenses

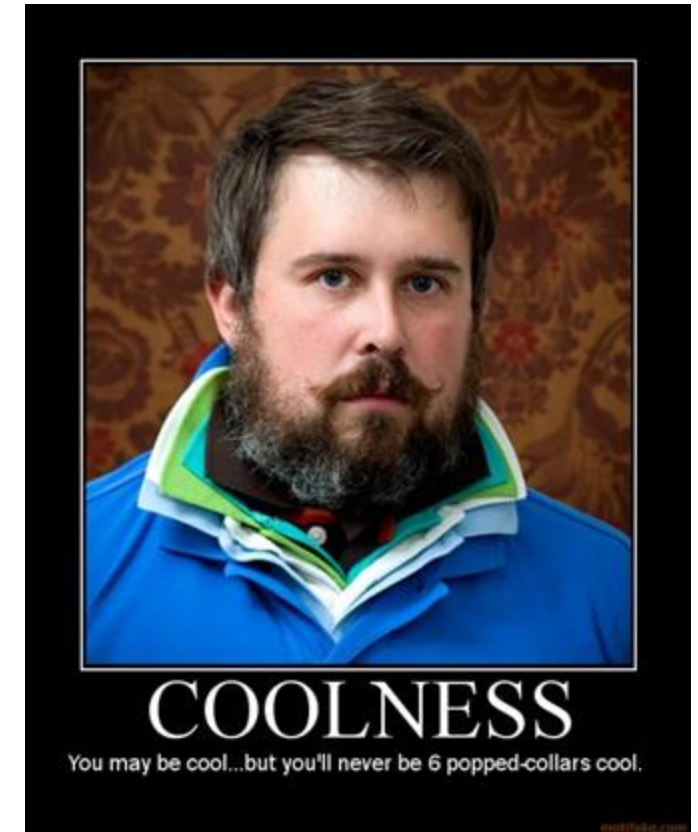


- *Lab: Bad Guy Defenses*



Lab: Bad Guy Defenses

- What OSes are they likely to use and why?
- What obfuscation techniques?
- What about persistence mechanisms?
- What about command and control (C2)?
- What about exfiltration techniques?
- Spend the next few moments and come up with a list...



Layers are not always
awesome.

You Will Be Exploited

- You should expect it. Anything less is denial...
- We focus far too much on prevention and not enough on detection and response
- Most current security technologies fail against these
 - Zero-day exploits
 - Phishing and SE
 - Advanced malware
 - Supply chain infiltration
 - Government backdoors (*sigh*)
- Expect the worst ... it's real



You might want to sit down for a while.

Segmentation

- Start segmenting your internal networks
 - All the way down to the desktop level
 - And between subnets
- Pass-the-Hash attacks have worked since 1997!
- Pass-the-Ticket and Security Access Token (SAT) impersonation have worked for years, too
- Make the assumption that you are going to get compromised
- Getting compromised is acceptable because it is going to happen
- What is unacceptable is an attacker persisting for months
- What is unacceptable is an attacker pivoting from one compromised system to the rest of the network in minutes
- Consider an “infected” VLAN

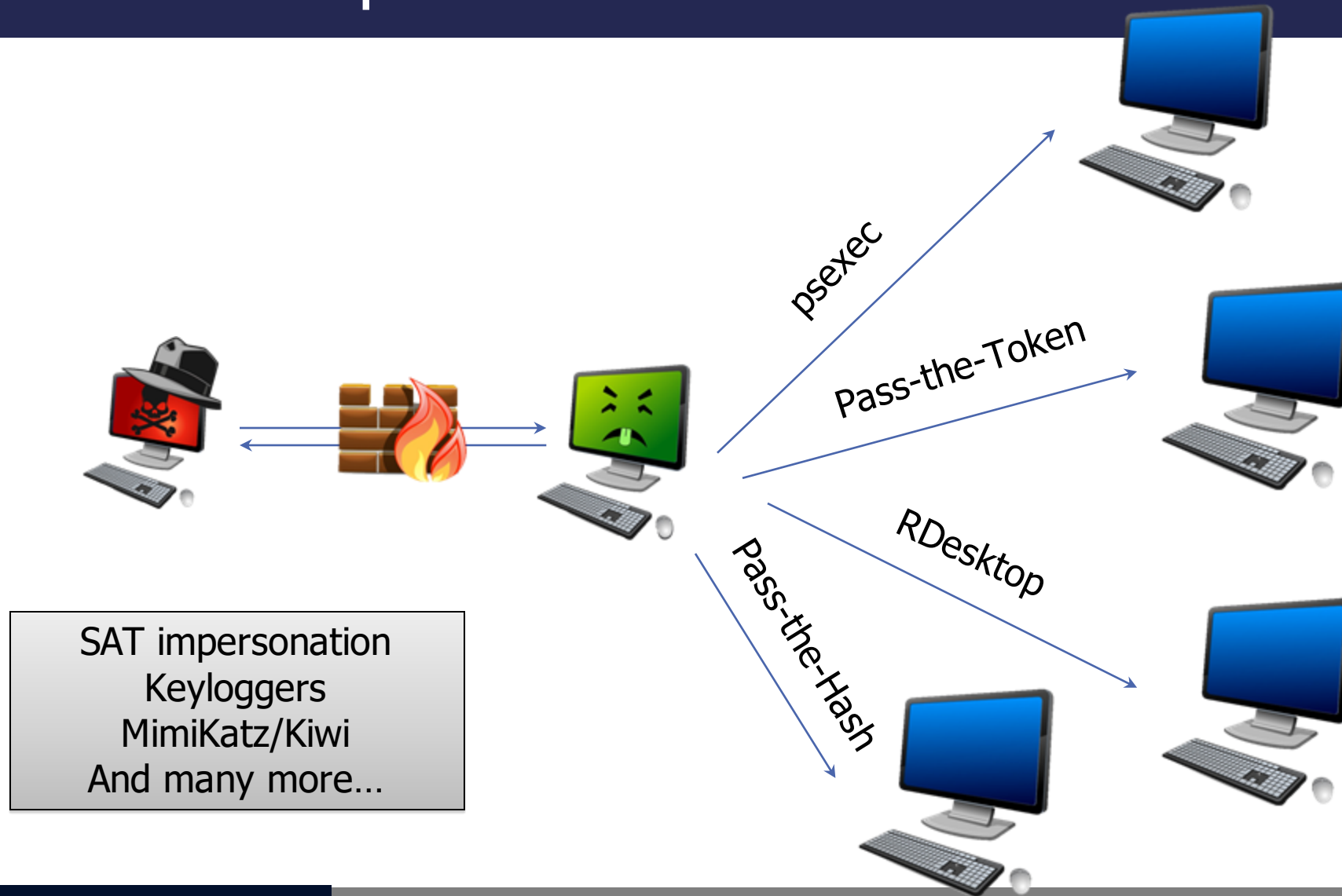
DTE0022	Isolation	Configure devices, systems, networks, etc. to contain activity and data in order to promote inspection or prevent expanding an engagement beyond desired limits.
DTE0025	Network Diversity	Use a diverse set of devices on the network to help establish the legitimacy of a decoy network.
DTE0026	Network Manipulation	Make changes to network properties and functions to achieve a desired effect.

Just Your Standard Exploit

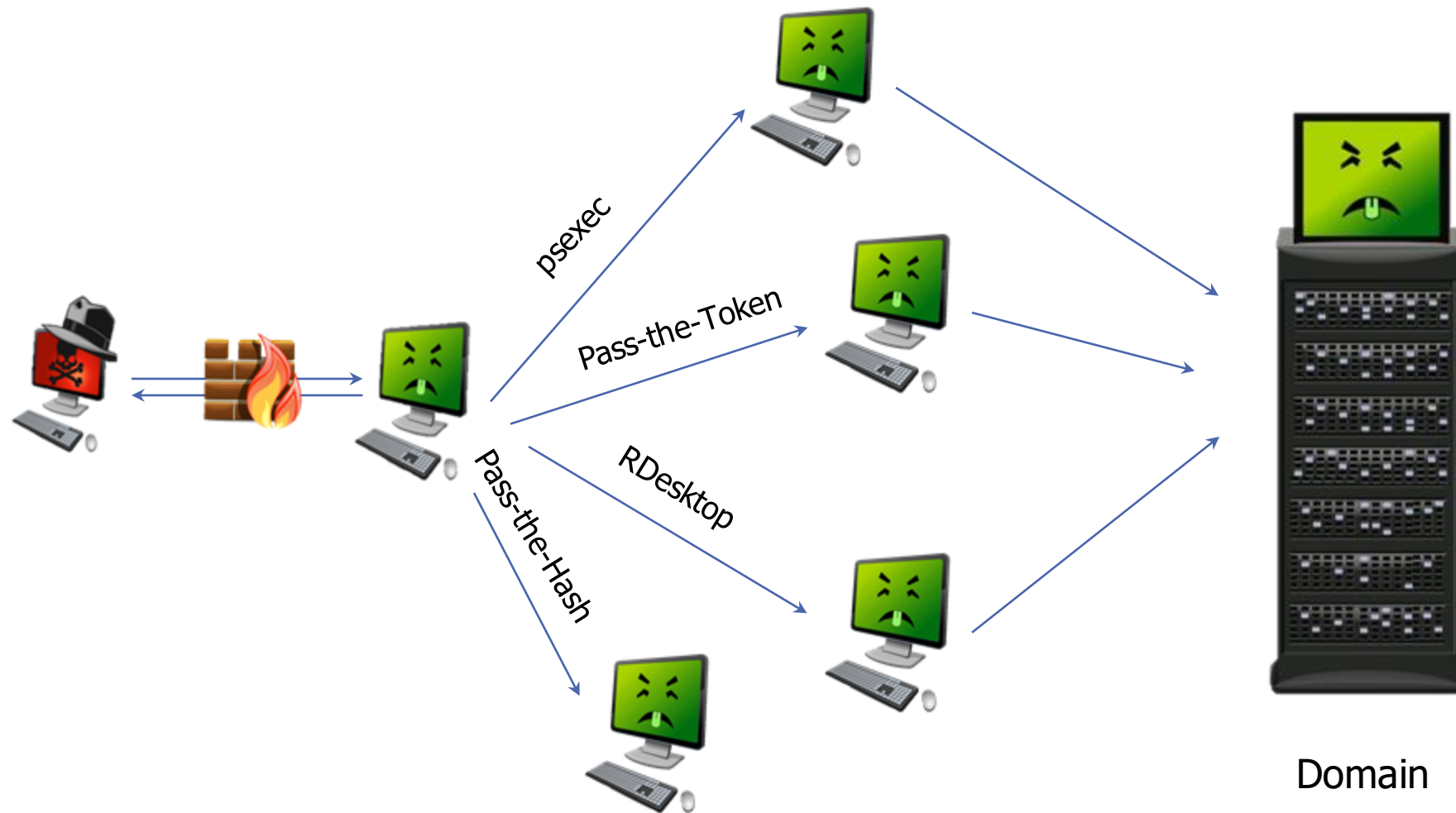


This is usually delivered as a client-side exploit or a drive-by download.

Will These Protocols Trip IDS Alerts?



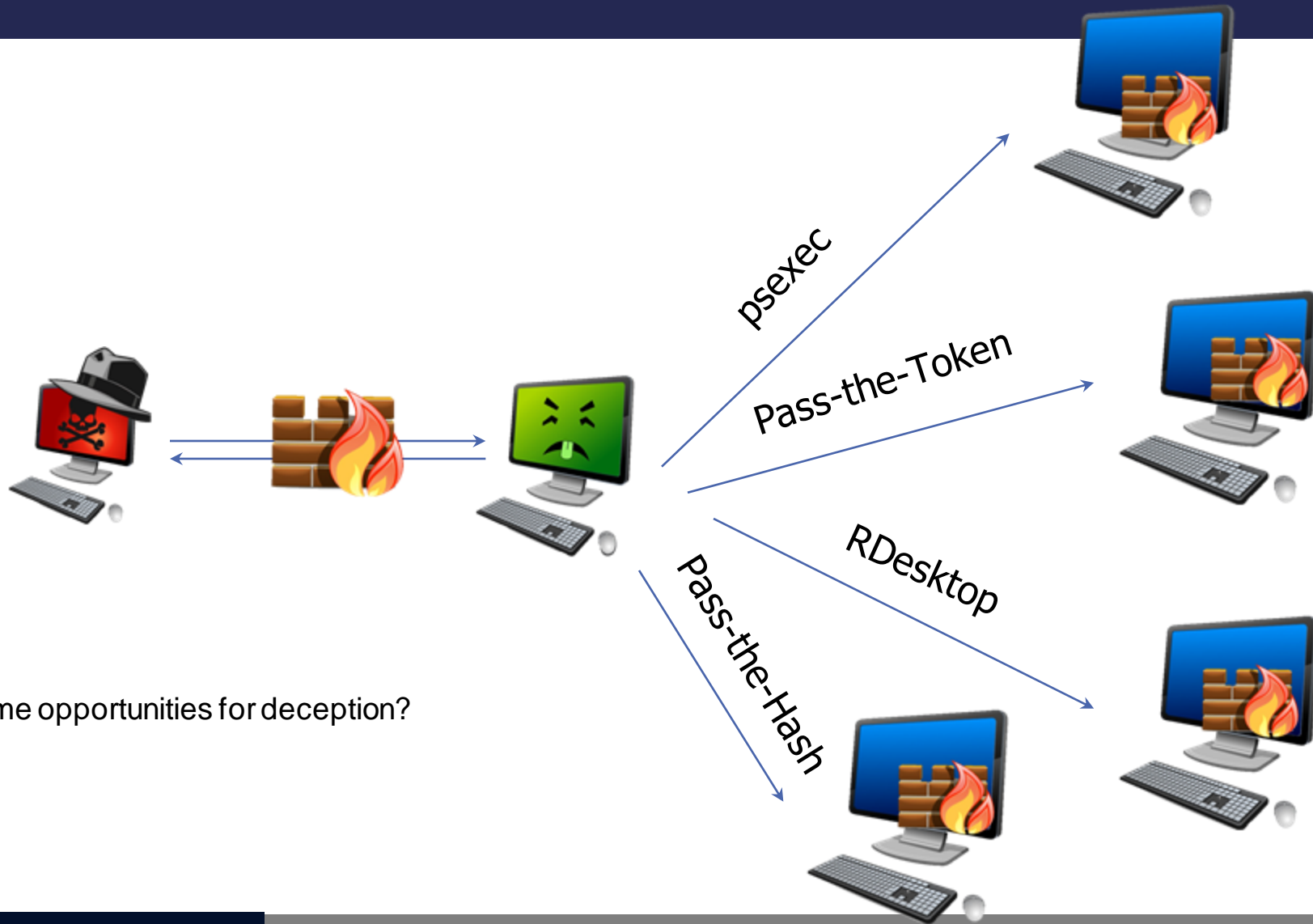
Most Likely They Will Not



Firewalls

- Treat the internal network as hostile
 - Because it is
- Set your internal system firewalls at the same level they would be at a coffee shop
 - All inbound traffic should be blocked and alerts should be generated
 - Exceptions for Admin networks
- Segment business units and/or organizational units
 - Why allow SMB RPC between subnets?
 - Contains the attacks even further than simple firewalls
- Many of the AV products have firewalls
- You can even use the built-in Windows firewall
 - If you are sadistic and desperate
- Private VLANs can work as well

Restriction of Lateral Movement



Where are some opportunities for deception?

Detecting an Insider

- If you cannot detect an insider, your network is not secure
 - Snowden
 - Attackers using valid/existing user credentials to move around a network
- Can you detect a user accessing 1000s of files?
- Can you detect an account that is accessing 100s of systems?
 - If not, you need to
- Future targeted attacks will use far less malware than now
- Would you be able to get proper attribution for an attacker who is on your system?
 - Word Web Bugs rock for this

Threat Emulation

- Don't just think of vulnerabilities as missing patches and misconfigurations on systems
- Think post exploitation
- What happens after an attacker gains access to a system
- There are a number of free tools that will automate parts of this process
- Currently, would take a bit of tuning and trial and error
- The collected data is invaluable

Open Source Tool Example: Caldera

The screenshot displays the Caldera web interface with a dark top navigation bar containing 'CALDERA', 'Threat', 'Networks', 'Operations', and 'Debug'. The main content area is divided into two panels. The left panel, titled 'Operation Overview', shows a 'test operation' in the 'running' phase, initiated by 'test adversary' on 'win7x01' at 11:00/2017, 8:38:57 PM. It lists 'Compromised Hosts' and 'Compromised Creds'. Below this is an 'Operation Graph' showing a network of five hosts: win7x01 (blue), win7x02, win7x03, win7x04, and win7x05 (all red). The right panel, titled 'Operation Details', features a 'Cancel Operation' button and tabs for 'Steps', 'Jobs', 'Artifacts', 'Cleanup Log', and 'BSF'. The 'Steps' tab is active, displaying a list of 15 tasks, including enumerating administrators, mounting network shares, copying implants, starting remote processes, and running mimikatz on various hosts.

Operation Overview

Status: **running** Phase: **operation** Action: **execute**

Operation: test operation
Start Time: 11/00/2017, 8:38:57 PM
Compromised Hosts 4

Adversary: test adversary
Starting Host: win7x01
Compromised Creds 1

Operation Graph

win7x01
win7x02
win7x03
win7x04
win7x05

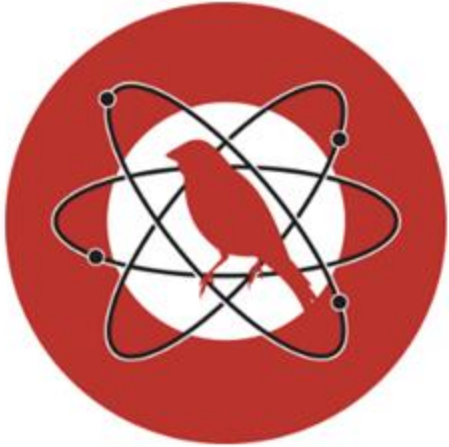
Operation Details

Cancel Operation

Steps Jobs Artifacts Cleanup Log BSF

- 1. Enumerating the Administrators group of win7x01.mountainpeak.local
- 1. Enumerating the Administrators group of win7x02.mountainpeak.local
- 1. Mounting win7x02.mountainpeak.local's C\$ network share on win7x01.mountainpeak.local with net use
- 1. Copying an implant from win7x01.mountainpeak.local to win7x02.mountainpeak.local
- 1. Starting a remote process on win7x02.mountainpeak.local using WMI
- 1. Running mimikatz to dump credentials on win7x02.mountainpeak.local
- 1. Mounting win7x03.mountainpeak.local's C\$ network share on win7x02.mountainpeak.local with net use
- 1. Copying an implant from win7x02.mountainpeak.local to win7x03.mountainpeak.local
- 1. Starting a remote process on win7x03.mountainpeak.local using WMI
- 1. Running mimikatz to dump credentials on win7x03.mountainpeak.local
- 1. Mounting win7x04.mountainpeak.local's C\$ network share on win7x03.mountainpeak.local with net use
- 1. Copying an implant from win7x03.mountainpeak.local to win7x04.mountainpeak.local
- 1. Starting a remote process on win7x04.mountainpeak.local using WMI
- 1. Running mimikatz to dump credentials on win7x04.mountainpeak.local

Open Source Tool Example: Atomic Red Team



Atomic Red Team

Execute All Attacks for a Given Technique

```
Invoke-AtomicTest T1117
```

Specify a Process Timeout

```
Invoke-AtomicTest T1117 -TimeoutSeconds 15
```

If the attack commands do not exit (return) within the specified `-TimeoutSeconds`, the process and its children will be forcefully terminated. The default value of `-TimeoutSeconds` is 120. This allows the `Invoke-AtomicTest` script to move on to the next test.

Execute All Tests

This is not recommended but you can execute all Atomic tests in your atomics folder with the following:

```
Invoke-AtomicTest All
```

Execute All Tests from a Specific Directory

Specify a custom path to your atomics folder, example `C:\AtomicRedTeam\atomics`

```
Invoke-AtomicTest All -PathToAtomicsFolder C:\AtomicRedTeam\atomics
```

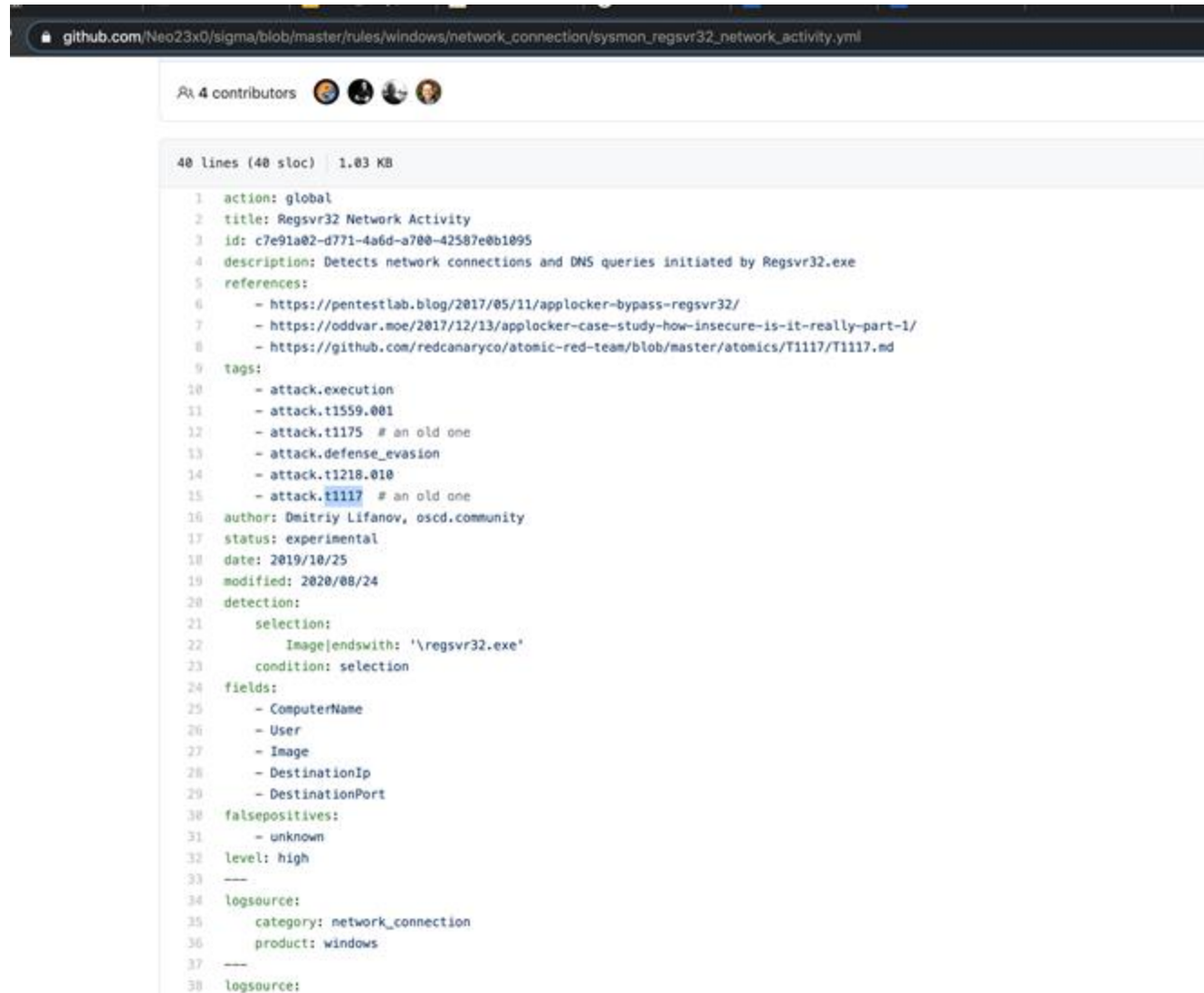
```

PS C:\AtomicRedTeam> Invoke-AtomicTest T1117 -TestNumbers 1 -ShowDetails
PathToAtomicsFolder = C:\AtomicRedTeam\atomics

[*****BEGIN TEST*****]
Technique: Regsvr32 T1117
Atomic Test Name: Regsvr32 local COM scriptlet execution
Atomic Test Number: 1
Description: Regsvr32.exe is a command-line program used to register and unregister OLE controls.
Upon execution, calc.exe will be launched.
Attack Commands:
Executor: command_prompt
ElevationRequired: False
Command:
regsvr32.exe /s /u /i:#{filename} scrobj.dll
Command (with inputs):
regsvr32.exe /s /u /i:C:\AtomicRedTeam\atomics\T1117\src\RegSvr32.sct scrobj.dll
Dependencies:
Description: Regsvr32.exe must exist on disk at specified location (C:\AtomicRedTeam\atomics\T1117\src\RegSvr32.sct)
Check Prereq Command:
if (Test-Path #{filename}) {exit 0} else {exit 1}
Check Prereq Command (with inputs):
if (Test-Path C:\AtomicRedTeam\atomics\T1117\src\RegSvr32.sct) {exit 0} else {exit 1}
Get Prereq Command:
New-Item -Type Directory (split-path #{filename}) -ErrorAction ignore | Out-Null
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomics/T1117/src/RegSvr32.sct" -OutFile " #{filename}"
Get Prereq Command (with inputs):
New-Item -Type Directory (split-path C:\AtomicRedTeam\atomics\T1117\src\RegSvr32.sct) -ErrorAction ignore | Out-Null
Invoke-WebRequest "https://github.com/redcanaryco/atomic-red-team/raw/master/atomics/T1117/src/RegSvr32.sct" -OutFile "C:\AtomicRedTeam\atomics\T1117\src\RegSvr32.sct"
[!!!!!!!END TEST!!!!!!]

```

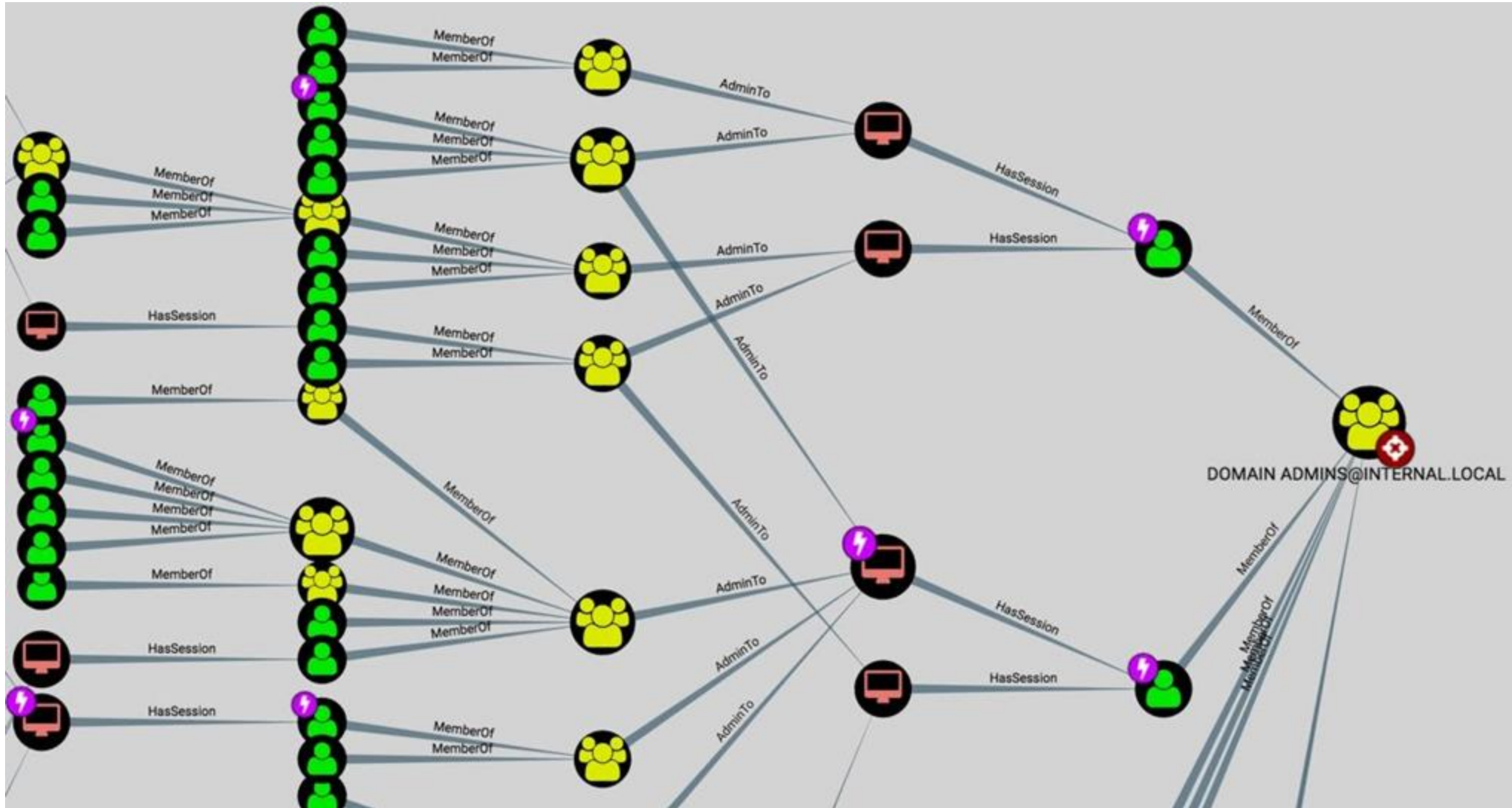
Sigma Detect



The screenshot shows a GitHub repository page for a Sigma rule. The URL in the browser is `github.com/Neo23x0/sigma/blob/master/rules/windows/network_connection/sysmon_regsvr32_network_activity.yml`. The repository has 4 contributors. The file is 40 lines (40 sloc) and 1.03 KB. The rule is a global action rule titled "Regsvr32 Network Activity" with ID `c7e91a02-d771-4a6d-a700-42587e0b1095`. The description states it detects network connections and DNS queries initiated by Regsvr32.exe. The rule includes references to several blogs and a GitHub repository, tags for attack execution, T1559, T1175, defense evasion, T1218, and T1117, an author of Dmitry Lifanov, an experimental status, a date of 2019/10/25, and a modification date of 2020/08/24. The detection rule selects for processes ending in `\regsvr32.exe` and includes fields for ComputerName, User, Image, DestinationIp, and DestinationPort. It also lists false positives as unknown and sets the level to high. The logsource is categorized as network_connection and the product as windows.

```
1 action: global
2 title: Regsvr32 Network Activity
3 id: c7e91a02-d771-4a6d-a700-42587e0b1095
4 description: Detects network connections and DNS queries initiated by Regsvr32.exe
5 references:
6   - https://pentestlab.blog/2017/05/11/aplocker-bypass-regsvr32/
7   - https://oddvar.moe/2017/12/13/aplocker-case-study-how-insecure-is-it-really-part-1/
8   - https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1117/T1117.md
9 tags:
10  - attack.execution
11  - attack.t1559.001
12  - attack.t1175 # an old one
13  - attack.defense_evasion
14  - attack.t1218.010
15  - attack.t1117 # an old one
16 author: Dmitry Lifanov, oscd.community
17 status: experimental
18 date: 2019/10/25
19 modified: 2020/08/24
20 detection:
21   selection:
22     Image|endswith: '\regsvr32.exe'
23   condition: selection
24 fields:
25   - ComputerName
26   - User
27   - Image
28   - DestinationIp
29   - DestinationPort
30 falsepositives:
31   - unknown
32 level: high
33 ---
34 logsource:
35   category: network_connection
36   product: windows
37 ---
38 logsource:
```


Open Source Tool Example: Bloodhound



Threat Emulation Warning

- One of the traps of the MITRE framework and threat emulation is we train our systems to detect specific attacks
- Most of the attacks in Atomic Red Team and MITRE are representations of classes of attacks
- We are seeing vendors simply detect those attacks
 - More on this later!
- A few modifications and you can easily bypass detection

Commercial Offerings

ATTACK IQ

XM CYBER



Getting Caught

Client malware detection and countermeasures			
HTTP viewstate covert channel - VSAgent; Port 443	2/1/2018 9:33	blocked	required authenticated proxy which is not compiled into client agent
DNSScat C2 channel; Port 53	2/1/2018 9:37	blocked	McAfee signature fired, and deleted malware
Metasploit HTTPS Meterpreter Shell code injected into memory via PowerShell; Port 443	1/31/2018 15:30	blocked	script would not seem to execute. No shell connection received
Metasploit TCP Meterpreter Shell code injected into memory via PowerShell (obfuscated with Unicorn); Port 443	2/1/2018 9:35	blocked	McAfee signature fired, and deleted malware
PowerShell Empire PowerShell code injected into memory; Port 443	2/1/2018 9:48	allowed	Command shell active
Raw malware EXE - Metasploit; Port 443; templated using write.exe	2/1/2018 9:56	allowed	Command shell active
Encoded malware EXE - Metasploit; Port 443; templated using write.exe	2/1/2018 9:57	allowed	Command shell active
MS-Office Document malicious macro; HTTPS port 443	2/1/2018 14:28	allowed	Command shell active
MS-Office Document malicious macro; TCP Port 8080	2/1/2018 14:34	blocked	McAfee Detected Malware
Cleartext communication with Netcat tool; Port 8443	2/1/2018 10:00	allowed	Anything that communicates with a TLS port such as 443 or 8443 is allowed through the perimeter without inspection
Metasploit Reverse TCP single stage EXE file.	2/1/2018 14:40	allowed	Command shell active
Metasploit Reverse TCP single stage Visual Basic file.	2/1/2018 14:39	blocked	McAfee Detected Malware
ICMP C2 Channel	2/1/2018 10:52	allowed	ICMP command shell established



Getting Caught 2

Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Execution	Collection	Exfiltration	Command and Control
Accessibility Features	Access Token Manipulation	Access Token Manipulation	Account Manipulation	Account Discovery	Application Deployment Software	Command-Line Interface	Audio Capture	Automated Exfiltration	Commonly Used Port
AppCert DLLs	Accessibility Features	Binary Padding	Brute Force	Application Window Discovery	Distributed Component Object Model	Dynamic Data Exchange	Automated Collection	Data Compressed	Communication Through Removable Media
AppInit DLLs	AppCert DLLs	Bypass User Account Control	Credential Dumping	File and Directory Discovery	Exploitation of Vulnerability	Execution through API	Browser Extensions	Data Encrypted	Connection Proxy
Application Shimming	AppInit DLLs	Code Signing	Credentials in Files	Network Service Scanning	Logon Scripts	Execution through Module Load	Clipboard Data	Data Transfer Size Limits	Custom Command and Control Protocol
Authentication Package	Application Shimming	Component Firmware	Exploitation of Vulnerability	Network Share Discovery	Pass the Hash	Graphical User Interface	Data Staged	Exfiltration Over Alternative Protocol	Custom Cryptographic Protocol
Bootkit	Bypass User Account Control	Component Object Model Hijacking	Forced Authentication	Peripheral Device Discovery	Pass the Ticket	InstallUtil	Data from Local System	Exfiltration Over Command and Control Channel	Data Encoding
Browser Extensions	DLL Search Order Hijacking	DLL Search Order Hijacking	Hooking	Permission Groups Discovery	Remote Desktop Protocol	LSASS Driver	Data from Network Shared Drive	Exfiltration Over Other Network Medium	Data Obfuscation
Change Default File Association	Exploitation of Vulnerability	DLL Side-Loading	Input Capture	Process Discovery	Remote File Copy	Mshta	Data from Removable Media	Exfiltration Over Physical Medium	Domain Fronting
Component Firmware	Extra Window Memory Injection	Deobfuscate/Decode Files or Information	LLMNR/NBT-NS Poisoning	Query Registry	Remote Services	PowerShell	Email Collection	Scheduled Transfer	Fallback Channels
Component Object Model Hijacking	File System Permissions Weakness	Disabling Security Tools	Network Sniffing	Remote System Discovery	Replication Through Removable Media	Regsvcs/Regasm	Input Capture		Multi-Stage Channels
Create Account	Hooking	Exploitation of Vulnerability	Password Filter DLL	Security Software Discovery	Shared Webroot	Regsvr32	Man in the Browser		Multi-hop Proxy
DLL Search Order Hijacking	Image File Execution Options Injection	Extra Window Memory Injection	Private Keys	System Information Discovery	Taint Shared Content	Rundll32	Screen Capture		Multiband Communication
External Remote Services	New Service	File Deletion	Replication Through Removable Media	System Network Configuration Discovery	Third-party Software	Scheduled Task	Video Capture		Multilayer Encryption
File System Permissions Weakness	Path Interception	File System Logical Offsets	Two-Factor Authentication Interception	System Network Connections Discovery	Windows Admin Shares	Scripting			Remote File Copy

Key Takeaways

- Moving from “Can we be hacked?”
 - To..
- “What can we detect?”
- We (finally) have a framework for this with MITRE
- We also have a large number of tools in their infancy to help automate this
- Start by finding gaps. Fill them. Move on.
- Start with the framework
- This is ACTIVE

steal this idea

Introductions and Standards

- *Lab: Playing with Advanced Backdoors*



Advanced C2: Lab Goals

- The goal of this lab is to understand how “advanced” backdoors operate
 - Beacons and obfuscation are key for a bad guy’s back door to persist
- We will look at a packet capture and decode the command and control data
- **We will use the ADHD VM for this lab**
- We will look at the packets and at RITA which will make it easier to detect
- The lab should take roughly 25 minutes

DTE0027	Network Monitoring	Monitor network traffic in order to detect adversary activity.
DTE0028	PCAP Collection	Collect full network traffic for future research and analysis.
DTE0031	Protocol Decoder	Use software designed to deobfuscate or decrypt adversary command and control (C2) or data exfiltration traffic.



Making it easier with RITA

Now! Follow the RITA Instructions on the class VM

- Now that we have looked at a problem backdoor, lets use a tool designed to make detection far easier!
- Open your Lab link on the ADHD VM and select the RITA section!



LAB: Conclusion

- How would IDS/IPS vendors write a signature for this type of traffic?
- Sure, they could write a signature for the specific Base64 string
 - But encryption and randomization would bypass that
- We could also implement Internet whitelisting
 - But in some organizations, this is simply not politically feasible
- This lab highlights just how hard it is to detect attackers when they are already in your network