Mandiant: Incident Response Manual

Logging Sources:

Host:

- Application
- System
- Host-based Firewall
- Powershell
- Antivirus
- Active Directory
- Software Inventory
- System Monitoring
- Critical Applications

Network:

- DHCP Leases
- DNS requests
- Firewall connections
- Proxy requests
- IDS/IPS alerts
- Email gateway
- Netflow
- Full Packet Capture
- VPN Logs

Hardening List:

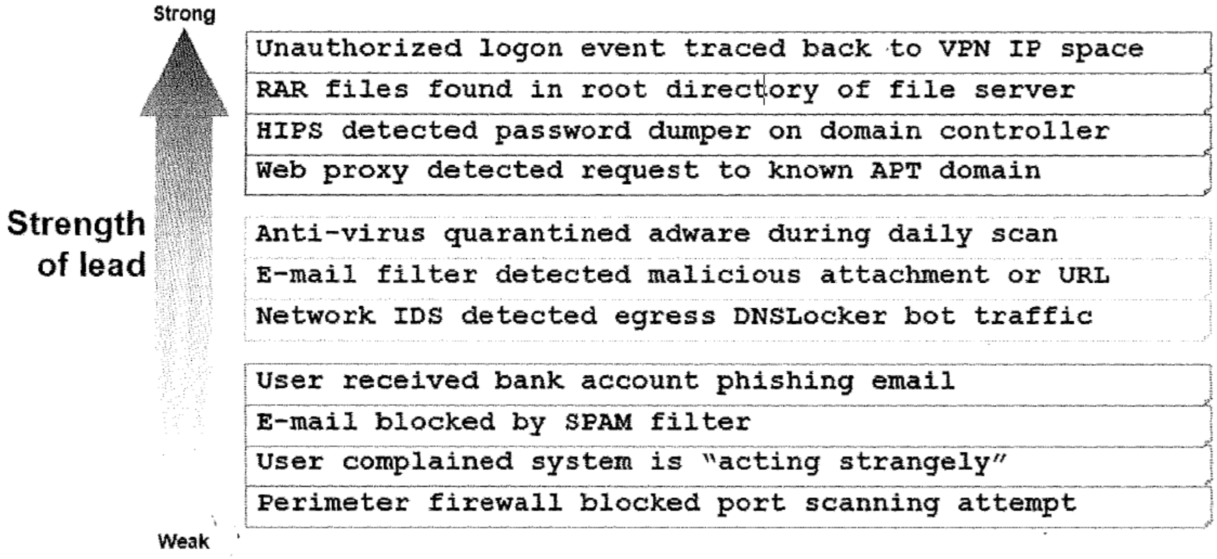
- Privilege management:
  - Remove administrator privileges from users
  - No shared accounts or password
  - Regular key rotation for privileged accounts
  - Enforce multi-factor authentication

- Network architecture:
  - Multi-tier segmentation
  - Block workstation-to-workstation communication
  - Limit workstation-to-server communication
  - Limit server-to-any communication
  - Bastion hosts to protect critical assets

- Host-Level security:
  - Routinely install system and application patches
  - Disable unnecessary features
  - Restrict access to privileged binaries
  - Untrusted binaries run in least privileged mode

- Enterprise-level security:
  - Regularly back up key systems
  - Restrict physical access to data centers
  - Encrypt data at rest and within databases

Prioritizing Investigative Leads



Unauthorized logon event traced back to VPN IP space
RAR files found in root directory of file server
HIPS detected password dumper on domain controller
Web proxy detected request to known APT domain

Anti-virus quarantined adware during daily scan
E-mail filter detected malicious attachment or URL
Network IDS detected egress DNSLocker bot traffic

User received bank account phishing email
E-mail blocked by SPAM filter
User complained system is "acting strangely"
Perimeter firewall blocked port scanning attempt

Sources of cyber security incidents:

1. Alert from an in-house technology (Reactive)
 - False positives need to be tuned out daily
 - Investigation playbooks should be developed per technique
2. Threat Hunting (Proactive)
 - Uses threat intelligence and attacker TTPs
 - Requires strong understanding of corporate network
3. External notification
 - Help desk notifies of anomaly
 - Law enforcement
 - Security researcher

Core Sources of Evidence:
o File System
o Windows Registry
o Event Logs
o Services
o Persistence Mechanisms
o Artifacts of Execution

Volatile evidence: stored in memory and lost when system is shut down. Examples: network connection states, process listing, process memory space.

Non-volatile evidence: stored on physical drives and preserved after shutdown. Examples: file system metadata, browser history, prefetch files, event logs, registry hives, scheduled tasks.

Incident Response Best Practices:

DO: Dump memory immediately, acquire live response data, acquire a forensic image of the system, maintain copies of all available backups in case of a roll-over, take snapshots of virtual machines, record the actions taken during response.

DO NOT: Shut down the system, connect to the system using Domain Admin credentials, delete attacker files or utilities, disable or terminate attacker processes, remediate before fully scoped, submit files to online services such as virustotal.

File System Analysis:

File metadata includes: file and directory names, timestamps, sizes, owner, access control lists, read-write-execute attributes. File contents includes: checksums, file type, strings.

What are "Services"?

- Background processes
- Automatically start upon boot or run manually
- Configured in the registry
- Can run as a stand-alone executable file
    - Image Path
- Can run as a DLL loaded by another executable file
    - Service DLL

Investigative questions:

- What services are currently installed?
    - Always check service binary checksum
- What is the state of each service?
- What modules do they load?

Persistence Mechanism: any access, action, or configuration change to a system that results in continuous access despite: system reboots and/or loss of credentials.

File locations for Auto-Run Keys: startup folders, browser helper objects, dlls search order hijacking, binary replacement, unquoted service paths.

Artifacts of Execution Sources:

- registry
    - shimcache (Tracks metadata for PE files and scripts)
    - amcache (Records significant file execution metadata)
    - userassist/muicache (Tracks files opened in Windows Explorer)
- prefetch (What applications previously executed, and when?)
- windows event logs (Detailed process auditing)
- wmi recently used apps (Execution history)

Performing Analysis:

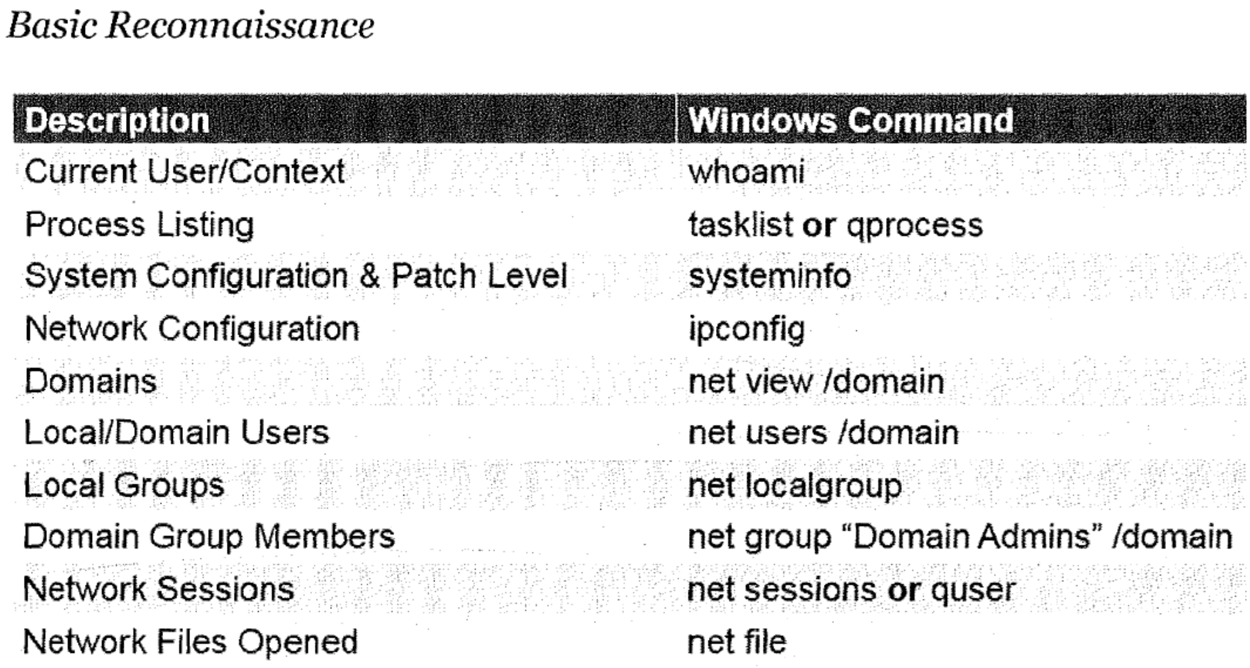
- Knowing what evidence/artifacts to collect
- Understanding the artifacts and their relation to attacker activity
- Telling a story
- Leveraging findings to identify additional systems

Timelining:

- Organizing artifacts based on temporal information
- Useful for identifying:
  - Order of attackers actions
  - Periods of interest
  - Effects of unrecoverable actions
  - Changes in tactics

Indicators of Compromise:

- Codify information on malware & utilities
- Supports various matching conditions
- Focused on host-based artifacts
- Re-usable and share-able



Basic Reconnaissance

Description	Windows Command
Current User/Context	whoami
Process Listing	tasklist **or** qprocess
System Configuration & Patch Level	systeminfo
Network Configuration	ipconfig
Domains	net view /domain
Local/Domain Users	net users /domain
Local Groups	net localgroup
Domain Group Members	net group “Domain Admins” /domain
Network Sessions	net sessions **or** quser
Network Files Opened	net file

Privilege Escalation: Becoming a user with higher capabilities.

Highest permission accounts:

- Local: NT AUTHORITY\SYSTEM
- Domain: A member of the Domain Administrators group

Achieved via:

- Built-in Windows Mechanisms
- Privilege escalation vulnerabilities
- Stealing credentials or access tokens

Local Privilege Escalation:

Primary ways:

- Service installation
- Scheduled Tasks
- User Access Control (UAC) bypass
- Shims, unquoted service paths, etc

Domain Privilege Escalation:

Windows Credentials - Kerberos:

- Default authentication protocol for Active Directory
- Mutual authentication protocol
- Client obtains "ticket" from Key Distribution Center
- Tickets presented to servers for authentication

Windows Credentials - Kerberos Attacks:

- DCSync
- Pass-the-ticket
- Kerberoasting
- SPN Scanning
- SYSVOL Passwords
- Skeleton Key attack
- Ticket forging

Lateral Movement:

- Logon Events
- Interactive Artifacts
- Remote Command Execution

Windows Logon Events:

- Type 2 - Interactive
  - Physical console
  - Screen sharing
  - RunAs
  - PsExec

- Type 10 - Remote Interactive
  - Remote Desktop/ Terminal Services

- Type 3 - Network
  - Access or transfer files
  - Native command line interaction
  - Interact with system services

Remote Command Execution:

- Executing processes on systems across the network
- Used for spreading backdoors and credential harvesting
  - Scheduled Tasks (Execute programs or scripts at defined intervals)

- PsExec (Sysinternals utility to remotely execute a program)
- Windows Management Instrumentation (WMI) (Allows for remote system management)
- PowerShell (Task automation and configuration management framework)

Alternate Remote Access:

- Increasingly prevalent in targeted attacks
  - VPN (Used to evade network monitoring)
  - DMZ Pivoting (Hop between· DMZ & other network segments)
  - Third Party Remote Access Utilities/ Solutions
- Attacker goes 'dark'
  - Suspends or removes primary backdoors

Completing the Mission - Data Theft:

- Data Collection and Aggregation
  - Compress and Encrypt Data
- Techniques for Data Theft
  - Backdoors
  - Built-In File Transfer Utilities (FTP/SFTP)
  - Cloud Services

Threat hunting is the process of applying our understanding of attackers and malware to raw data in order to find evil. Objective: To find previously undiscovered attacks related to current incidents, or related to threats targeting the enterprise's industry, geographical locations, applications, etc

Why is Threat Hunting Important?

- Skilled adversaries know how to avoid traditional detection.
- Threat hunting yields new detection methods.
- Knowing your environment is critical for effective response.

Threat Hunting and Analysis:

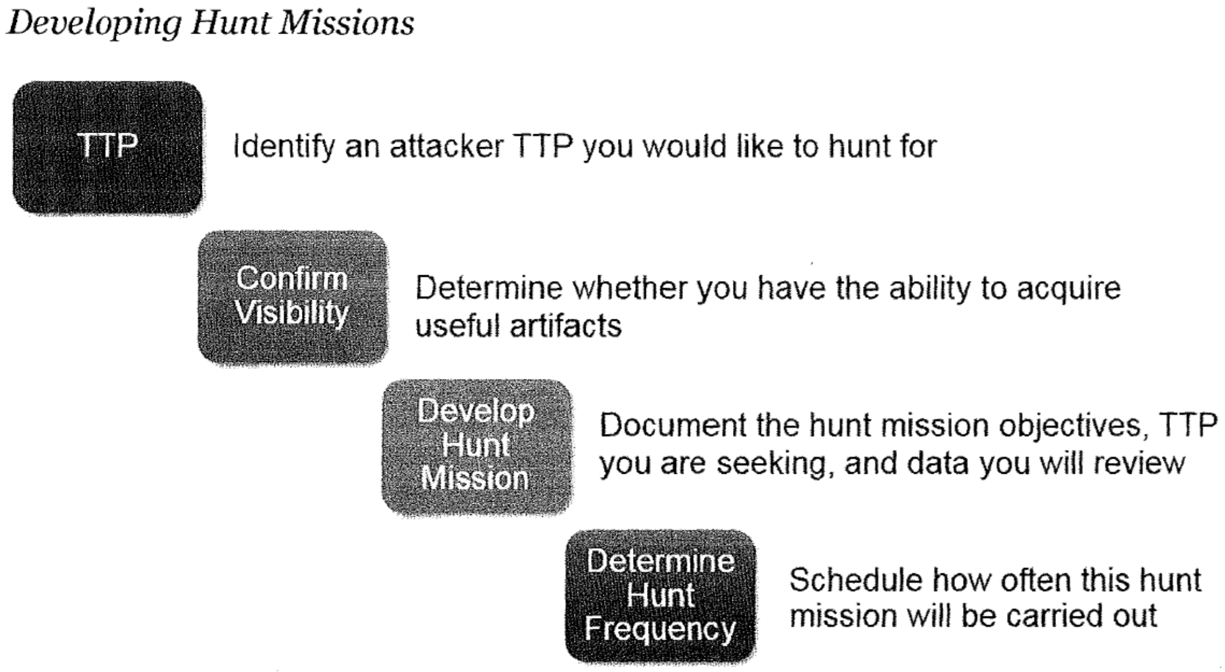
- Indicator-based hunting is faster, enables a higher confidence of evil
- Anomaly-based hunting requires expertise and time

What to look for?

- Specific indicators (known bad)
- Anomalies (unknown bad)

Threat Hunting Program Objectives:

- Develop repeatable and consistent hunting processes
- Use tools and automation to minimize manual activities
- Leverage analytics and visualizations to help with anomaly detection
- Review datasets that are not feasible for automated alerting (High-volume IDS alerts)
- Enhance detection mechanisms
- Hunt Historically, Alert Forward
- Goal: Fine-tune your hunts well enough to become alerts



Developing Hunt Missions

TTP — Identify an attacker TTP you would like to hunt for

Confirm Visibility — Determine whether you have the ability to acquire useful artifacts

Develop Hunt Mission — Document the hunt mission objectives, TTP you are seeking, and data you will review

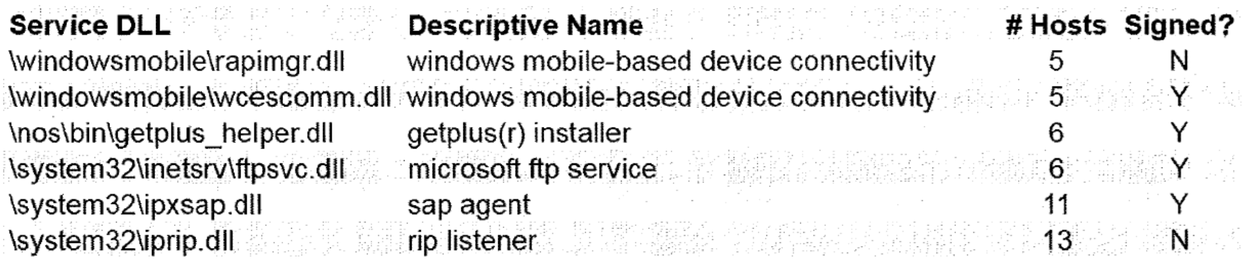
Determine Hunt Frequency — Schedule how often this hunt mission will be carried out

7 Threat Hunting Examples

1. Windows Services
2. Process Analysis
3. Scheduled Task Analysis
4. Web Traffic Analysis
5. Network Anomalies
6. Remote Access Anomalies
7. Suspicious Executables

Windows Services Analysis:

- 12 least-common Service DLLs among ~10,000 hosts
    - Eliminate files with valid digital signatures
    - Acquire suspicious files as necessary



Service DLL	Descriptive Name	# Hosts	Signed?
\windowsmobile\rapimgr.dll	windows mobile-based device connectivity	5	N
\windowsmobile\wcescomm.dll	windows mobile-based device connectivity	5	Y
\nos\bin\getplus_helper.dll	getplus(r) installer	6	Y
\system32\inetsrv\ftpsvc.dll	microsoft ftp service	6	Y
\system32\ipxsap.dll	sap agent	11	Y
\system32\iprip.dll	rip listener	13	N

Process Analysis:

- Parent-Process pairings that follow common attacker methodologies
    - Example: Winword.exe launching rundll32.exe
- Frequency analysis/stacking to identify atypical executables
- Run Command Line Analysis for commonly abused attacker binaries
- Search for encoded, obfuscated or obscured parameters

Scheduled Task Analysis:

- Unnamed Scheduled Task analysis (at jobs)
- Abnormal task naming conventions
- Scheduled task frequency analysis
- Abnormal task functions
  - Scripts (BAT, CMD, WSF)
  - Remote administration tools, atypical binaries

Web Traffic Analysis:

- Abnormal outbound traffic, blocked traffic, IoC triggers
- Correlation is your strongest tool
- Site registration date, known hosting providers

Network Anomalies Analysis:

- Failed traffic analysis
  - Outbound denied on ports 80, 443, 53
- Abnormal traffic patterns
  - Outbound non-DNS allows on port 53
- Abnormal protocol usage
  - Protocol usage on unusual ports (SSH on 21)
- Visualization & Data Analytics makes it easier to investigate outliers

Remote Access Anomalies Analysis:

- Geolocation / Geofeasibility analysis
  - Correlate distance between geolocation points and login times
- Remote application analysis
  - Investigate long sessions with only access to Explorer/CMD
- Session length analysis
  - Investigate abnormally long sessions (>24 hours)

Suspicious Executables Analysis:

- Executable file frequency analysis
  - Randomly generated filenames, abnormal filenames
- Abnormal executable locations
  - Temp Directories (C:Windows\Temp, C:\temp, C:\tmp}
  - AppData Directories (C:\Users\*\appdata)
- MD5 indexing and lookup
  - System folder locations (system32, C:\Windows)