OS Analysis with Volatility

Detect and Respond with Volatility

Tim Coakley

Senior Security Solutions Architect

https://www.linkedin.com/in/timcoakley/

VOLATILITY

VOLATILITY

Creator: Aaron Walters / Volatility Foundation

Volatility is an open-source memory forensics framework for incident responders, forensic practitioners and malware analysts.

Simple to operate command line tool

Supports broad range of memory images

Opensource License

Available to download on GitHub

Supports memory format conversions

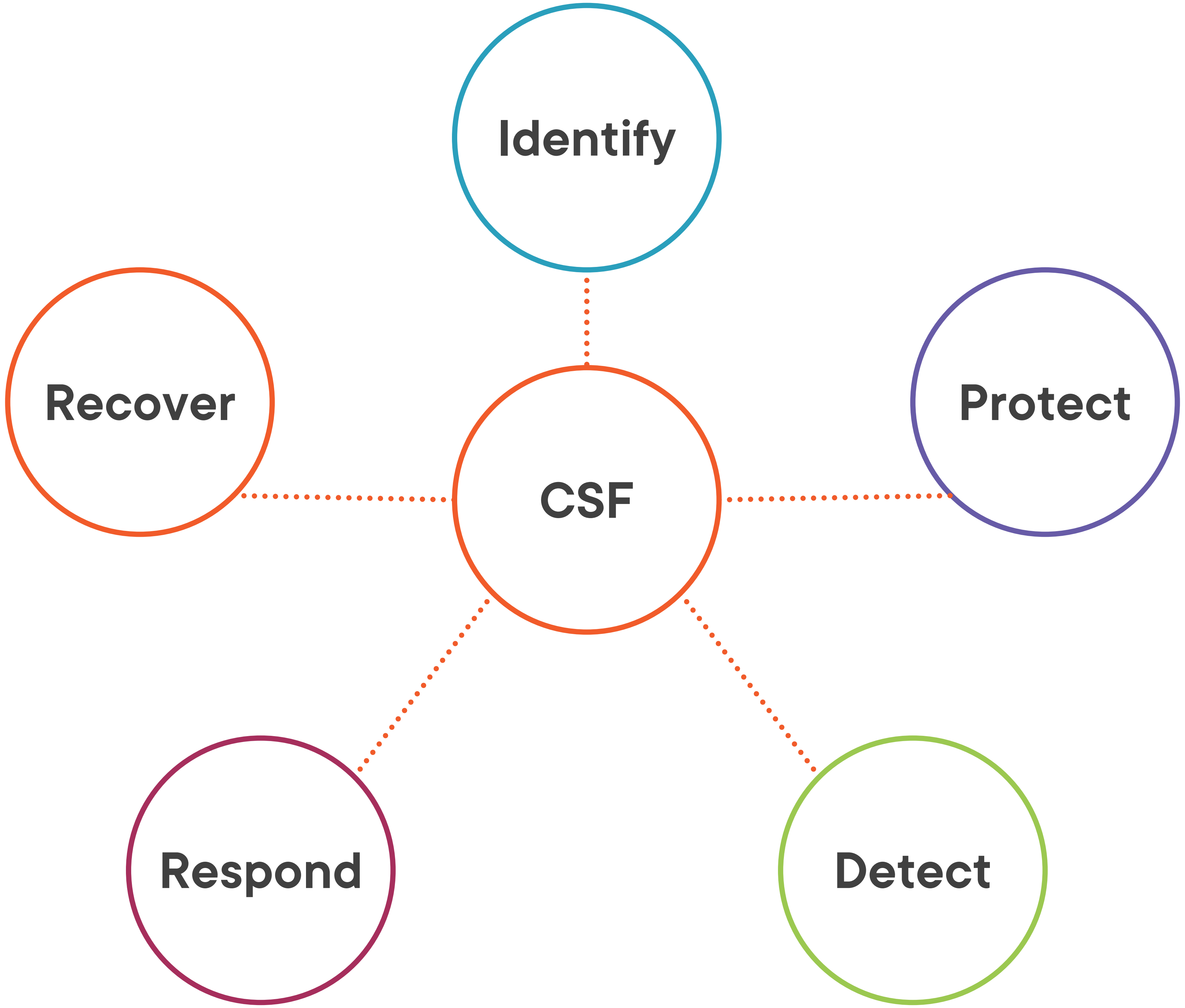
Includes optional plugins

VOLATILITY

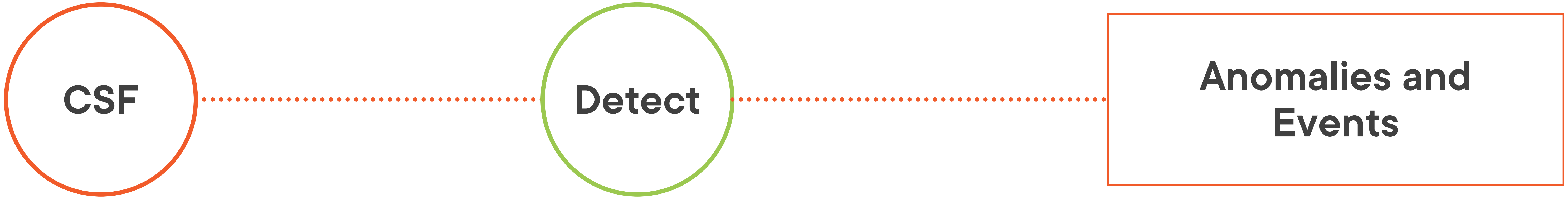
Includes option to create customized plugins

Flexible reporting onscreen or to file

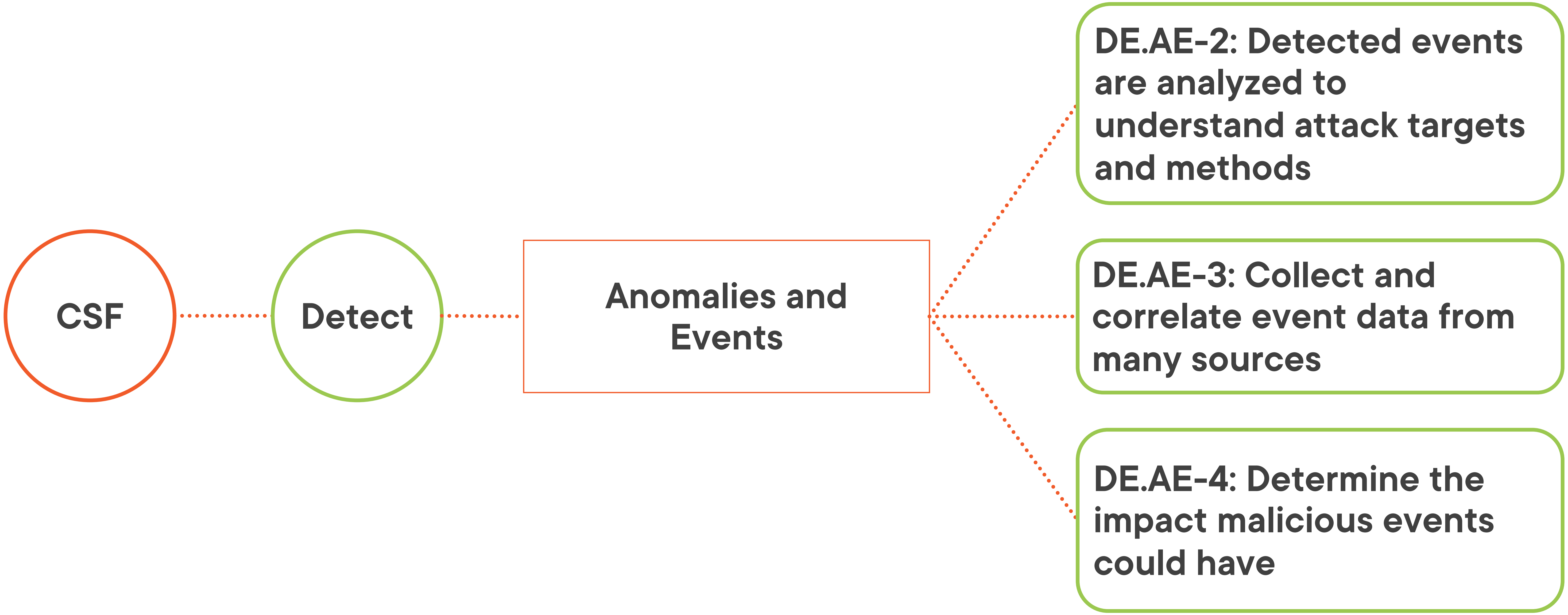
NIST Cybersecurity Framework



NIST Cybersecurity Framework



NIST Cybersecurity Framework



CSF

Detect

Anomalies and Events

DE.AE-2: Detected events are analyzed to understand attack targets and methods

DE.AE-3: Collect and correlate event data from many sources

DE.AE-4: Determine the impact malicious events could have

MITRE ATT&CK

Data Analysis Type

- Network Analysis
- OS Analysis
- Application Analysis
- Infrastructure Analysis
- File Analysis
- Threat Intelligence
- Incident Management

MITRE ATT&CK

Data Type

- Network Analysis
- OS Analysis
- Application Analysis
- Infrastructure Analysis
- File Analysis
- Threat Intelligence
- Incident Management

T1059:
Command and Scripting Interpreter

T1055:
Process Injection

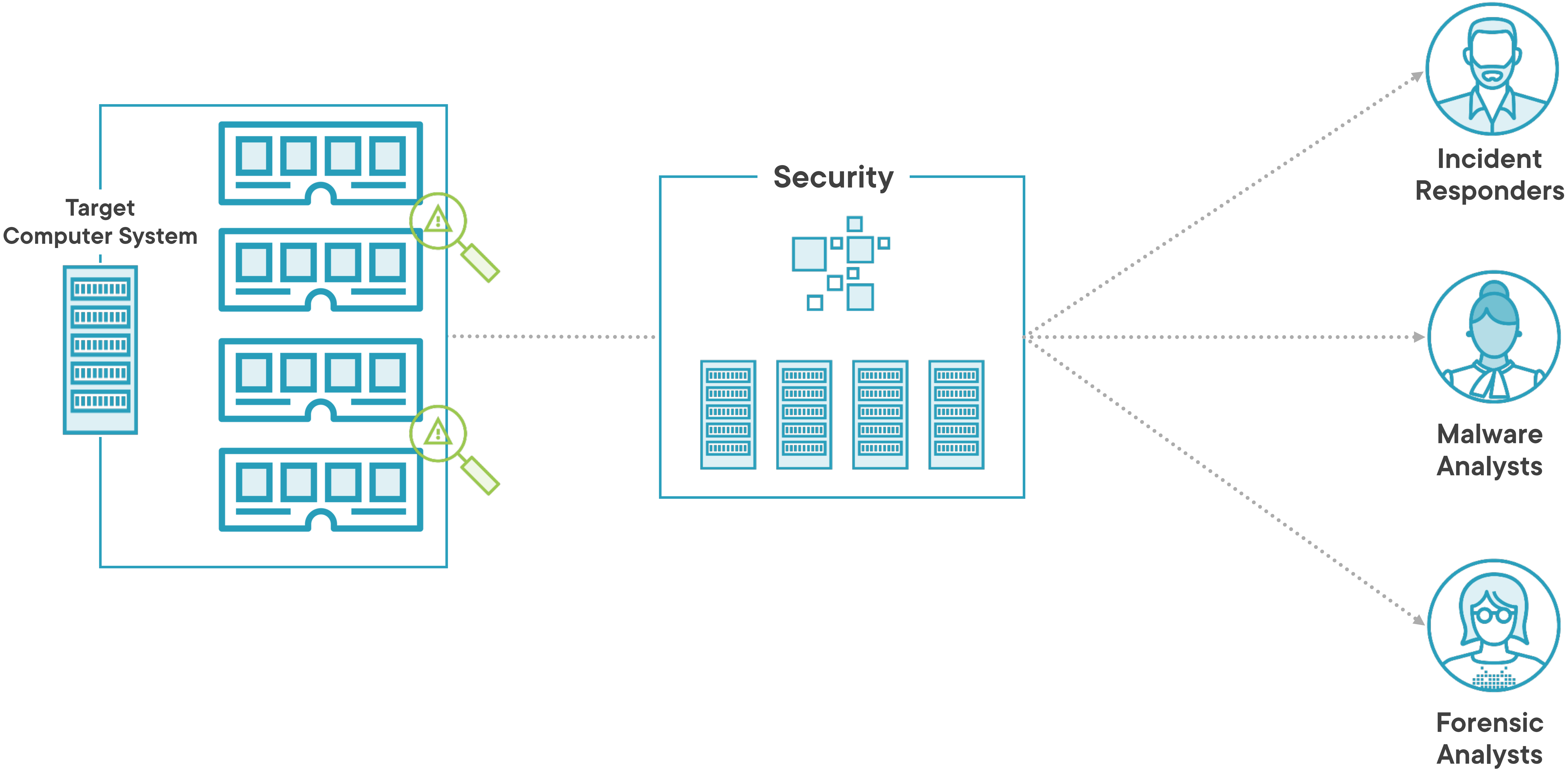
MITRE SHIELD

T1059:
Command and Scripting Interpreter

DTE0034 – System Activity Monitoring: a defender can detect the presence of an adversary by monitoring for processes that are created by commands and/or scripts that they execute on a system. (DUC0033)

T1055:
Process Injection

DTE0032 – Security Controls: a defender can block execution of untrusted software. (DUC0048)


Target
Computer System
Security
Incident
Responders
Malware
Analysts
Forensic
Analysts

Incident Analysis and Detection

Malware Analysis

Detailed Investigation (forensic)

Volatility needs an image taken of memory to be able to perform analysis.

Installation details are available on the Volatility GitHub page.

KEY TAKEAWAYS

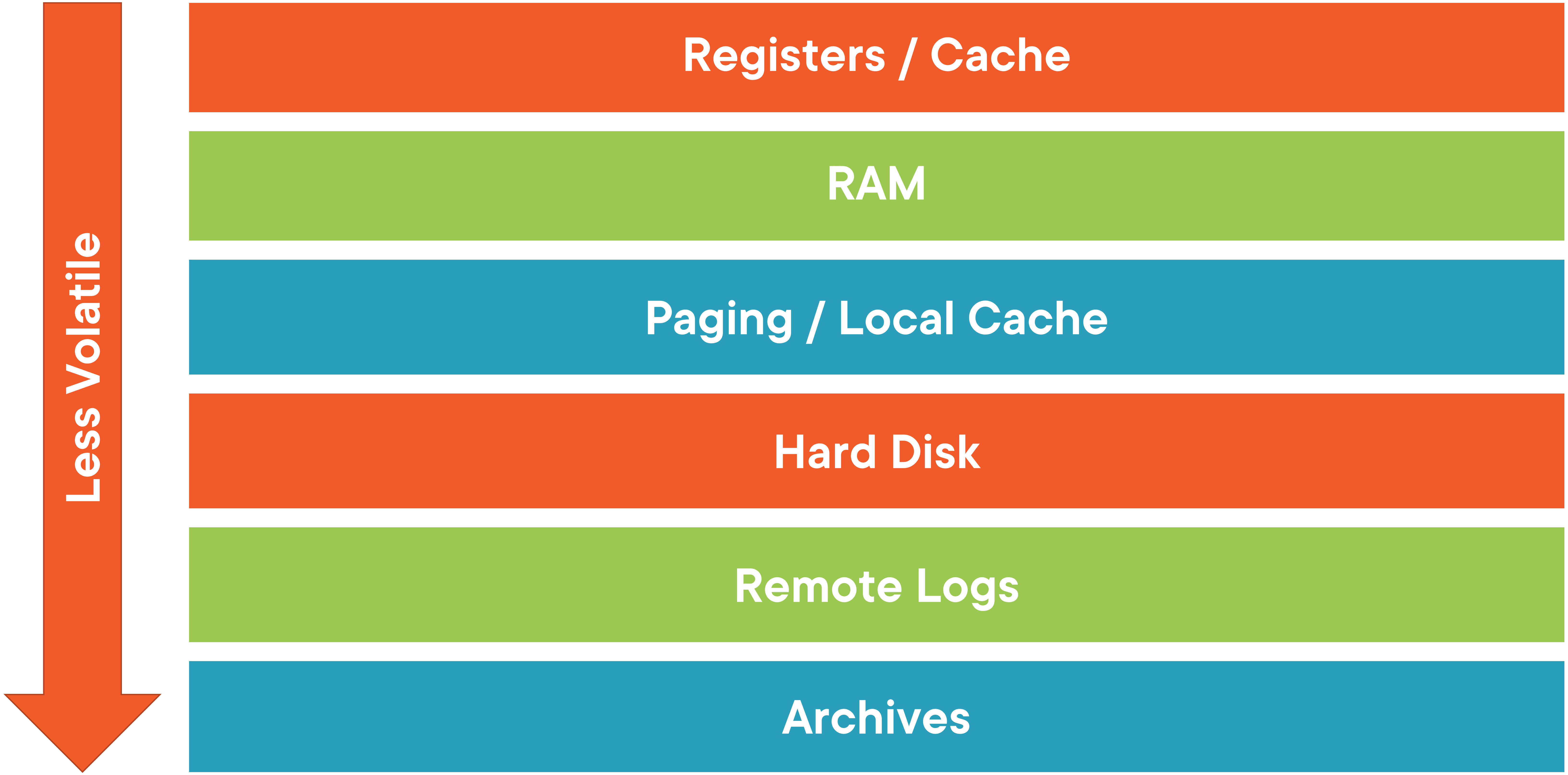
Memory Analysis

Develop ability to analyze volatile memory

Attacker Motivations

Understand attacker tools and techniques

Order of Volatility



Less Volatile

Registers / Cache

RAM

Paging / Local Cache

Hard Disk

Remote Logs

Archives

Command History - Evidential Benefits

Evidence of User Attribution

Evidence of Lateral Movement

Evidence of Privilege Escalation

Approved vs Unapproved

Approved

Authorized personnel who have a genuine justification to use terminal commands

Unapproved

Unauthorized personnel using terminal commands, difficult to monitor

Common Volatility Plug-ins

pslist

connections

pstree

process

cmdscan

consoles

Approved User Roles

Administrators

Maintaining systems and services

Developers and Testers

Writing code and installing software

Other

Contractors, third party personnel

Unapproved User Roles

Insider Threat

Accidental, non-malicious users

Insider Threat

Malicious, intention damage or steal information

External Threat

Opportunistic, targeted, exploitative

Command History Plug-ins

cmdscan

Scan for console commands entered by user

consoles

Scan for console commands including screen buffer

Demo

Triage Volatile Memory

Review for command line activity

When to perform memory analysis?

Memory analysis is both time and labor intensive requiring skilled cyber security personnel, so perform only when necessary

Alert

An alert generated from security software may initiate analysis

External Request

A request from another team or colleague may initiate analysis

Investigation

Investigation steps may vary based on the initial request

What formats are supported?

RAW

RAM from physical machines

VMWare

Saved state and snapshots

Virtualbox

Coredumps and state files

Hibernation

Windows hibernation files

More

Numerous other supported formats...

Example Security Incident

Security Log Alert generated 3 May 2021 19:23 hours

Internal Server with IP address 10.11.1.12

Connecting to external IP address 54.93.101.71

Example Security Incident…continued

Memory Image of the server is captured and saved as a file

Analyze the memory image for suspicious activity

Provide security recommendations

Demo

Analyze a suspect memory image

General Remediation Actions

Review Access

Edit and remove access to console

Disable Credentials

Disable accounts so they cannot be used

Secure Build

Secure workstations & use approved software

Rotate Credentials

Changing a password from old to new

Incident Response

Update playbooks for volatile analysis

Reducing Risk

Security Awareness

Strong IAM Controls

Monitoring

Demo

Mitigate against future attacks