



SANS

www.sans.org

SECURITY 511
CONTINUOUS MONITORING
AND SECURITY OPERATIONS

511.4

Endpoint Security Architecture

The right security training for your staff, at the right time, in the right location.
<https://t.me/learningnets>

Copyright © 2015, The SANS Institute. All rights reserved. The entire contents of this publication are the property of the SANS Institute.

IMPORTANT-READ CAREFULLY:

This Courseware License Agreement ("CLA") is a legal agreement between you (either an individual or a single entity; henceforth User) and the SANS Institute for the personal, non-transferable use of this courseware. User agrees that the CLA is the complete and exclusive statement of agreement between The SANS Institute and you and that this CLA supersedes any oral or written proposal, agreement or other communication relating to the subject matter of this CLA. If any provision of this CLA is declared unenforceable in any jurisdiction, then such provision shall be deemed to be severable from this CLA and shall not affect the remainder thereof. An amendment or addendum to this CLA may accompany this courseware. BY ACCEPTING THIS COURSEWARE YOU AGREE TO BE BOUND BY THE TERMS OF THIS CLA. IF YOU DO NOT AGREE YOU MAY RETURN IT TO THE SANS INSTITUTE FOR A FULL REFUND, IF APPLICABLE. The SANS Institute hereby grants User a non-exclusive license to use the material contained in this courseware subject to the terms of this agreement. User may not copy, reproduce, re-publish, distribute, display, modify or create derivative works based upon all or any portion of this publication in any medium whether printed, electronic or otherwise, for any purpose without the express written consent of the SANS Institute. Additionally, user may not sell, rent, lease, trade, or otherwise transfer the courseware in any way, shape, or form without the express written consent of the SANS Institute.

The SANS Institute reserves the right to terminate the above lease at any time. Upon termination of the lease, user is obligated to return all materials covered by the lease within a reasonable amount of time.

SANS acknowledges that any and all software and/or tools presented in this courseware are the sole property of their respective trademark/registered/copyright owners.

AirDrop, AirPort, AirPort Time Capsule, Apple, Apple Remote Desktop, Apple TV, App Nap, Back to My Mac, Boot Camp, Cocoa, FaceTime, FileVault, Finder, FireWire, FireWire logo, iCal, iChat, iLife, iMac, iMessage, iPad, iPad Air, iPad Mini, iPhone, iPhoto, iPod, iPod classic, iPod shuffle, iPod nano, iPod touch, iTunes, iTunes logo, iWork, Keychain, Keynote, Mac, Mac Logo, MacBook, MacBook Air, MacBook Pro, Macintosh, Mac OS, Mac Pro, Numbers, OS X, Pages, Passbook, Retina, Safari, Siri, Spaces, Spotlight, There's an app for that, Time Capsule, Time Machine, Touch ID, Xcode, Xserve, App Store, and iCloud are registered trademarks of Apple Inc.

Sec511_4_A13_01

Continuous Monitoring and Security Operations Endpoint Security Architecture

SANS Security 511.4
Seth Misenar (GSE #28) & Eric Conrad (GSE #13)

© 2015, Seth Misenar and Eric Conrad
All Rights Reserved
Version A13_01

Continuous Monitoring and Security Operations

I

Welcome to 511.4 Endpoint Security Architecture.

Course Outline

- Day 1: Current State Assessment, SOCs, and Security Architecture
- Day 2: Network Security Architecture
- Day 3: Network Security Monitoring
- **Day 4: Endpoint Security Architecture**
- Day 5: Automation and Continuous Security Monitoring
- Day 6: Capstone: Design, Detect, Defend

Continuous Monitoring and Security Operations

2

Course Outline

Now we turn our attention to Endpoint Security Architecture.

511.4 Table of Contents (1)

	Slide #
• Endpoint Security Architecture Overview	5
• Windows Endpoints.....	9
• Patching.....	14
• Secure Baseline Configuration.....	30
• EMET.....	43
• Sysmon.....	47
• Exercise: Sysmon	58
• Application Whitelisting.....	60
• Administrative Accounts.....	93
• Privilege Monitoring.....	106
• Exercise: AutoRuns	116
• Privilege Reduction.....	118
• Authentication.....	131

Continuous Monitoring and Security Operations

3

511.4 Table of Contents (1)

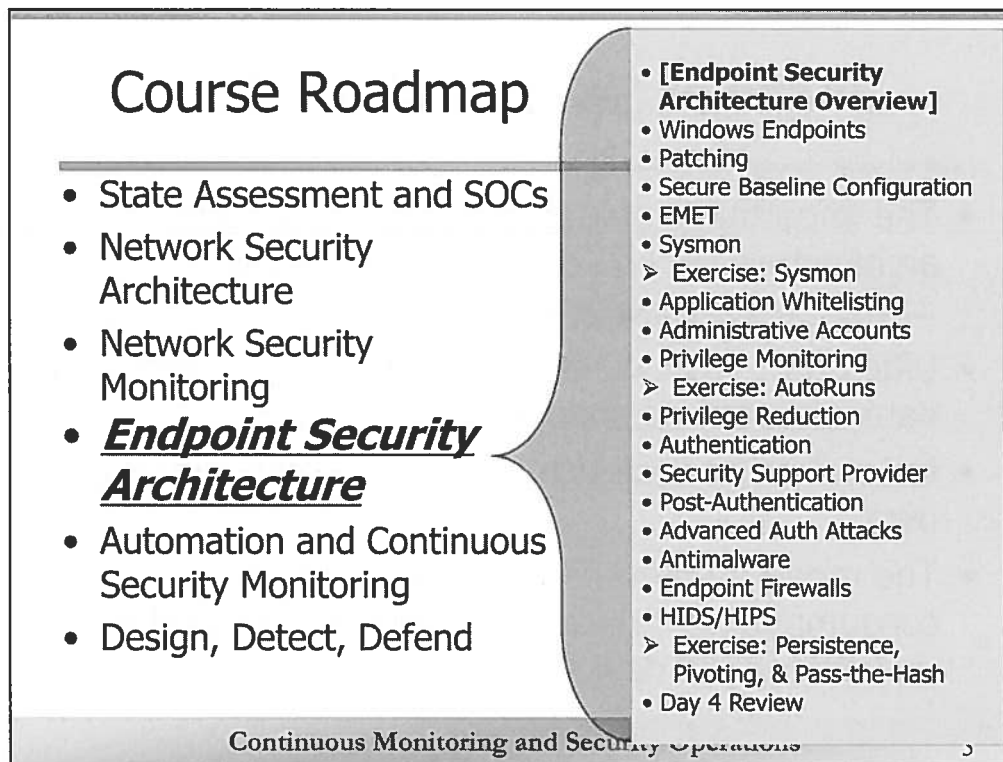
This table of contents outlines our plan for 511.4.

511.4 Table of Contents (2)

	Slide #
• Security Support Provider (SSP)	142
• Post-Authentication.....	151
• Advanced Authentication Attacks.....	158
• Antimalware.....	170
• Endpoint Firewalls.....	173
• Host Intrusion Detection/Prevention Systems.....	178
• Exercise: Persistence/Pivoting/Pass-the-Hash	190
• 511.4 Summary.....	192
• Immersive Cyber Challenges (NETWARS)	195

511.4 Table of Contents (2)

This table of contents outlines our plan for 511.4.



Course Roadmap

Our first section presents an overview of Endpoint Security Architecture.

Endpoint Security Architecture

- The importance of a strong network security architecture (511.2) cannot be overstated
 - And it is supportive of endpoint security
- Ultimately, what we are typically trying to secure is data, situated on endpoints
- Naturally, the easiest to secure are endpoints we own
- The modern enterprise must account for consumption of critical data from unmanaged or undermanaged devices

Continuous Monitoring and Security Operations

6

Endpoint Security Architecture

Adversaries' goals are focused largely on data, which is necessarily situated on endpoints. Conceptually, protecting an individual endpoint is far simpler than providing protection for multiple disparate devices, as we do with network protections. However, in practice, the difficulty of successfully employing robust security practices on endpoints proves difficult due to the volume of the endpoints that need to be protected.

One difficulty we routinely encounter in the modern enterprise is having to provide meaningful security to endpoints that are unmanaged, or, at the very least, undermanaged devices. Mobile devices obviously come to mind on this front.

{CSC} Critical Security Controls

- To ensure the validity of our approach we attempt to track back to the Critical Security Controls
- A large number of the controls are relevant to this day's material, and will be called out specifically
- Additionally all of the First Five Quick Wins noted in the CSC are directly related to today's material

Continuous Monitoring and Security Operations

7

Critical Security Controls

The 20 Critical Security Controls for Effective Cyber Defense serve as the major underlying sanity check for what is covered and why we cover it in this course. Today's material on endpoint security architecture will help to ensure our individual assets are defensible, and again the Critical Security Controls serve as a nice backdrop to ensure that we are focused on the most important and relevant security aspects.

We will specifically call out when a relevant CSC is discussed overtly in the course content. Most important will be the First Five Quick Wins.

First Five Quick Wins

- Application Whitelisting (CSC2)
- Use common, secure configs (CSC3)
- Patch applications in 48h (CSC4)
- Patch systems in 48h (CSC4)
- Reduce the number of users with administrative privileges (CSC3, CSC12)

Continuous Monitoring and Security Operations

8

First Five Quick Wins

Among the 20 Critical Security Controls, and the associated individual recommendations that comprise them, there are five particular recommendations that prove so important that they are called out specifically as the First Five Quick Wins.

The intention is that these 5 provide for some of the most significant security wins an organization can achieve. Implementation of simply these 5 will afford an organization a much more robust security posture. Given the previously discussed emphasis on compromise of data on endpoint systems, it should be unsurprising that the quick wins are most relevant to endpoint security architecture.

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- **[Windows Endpoints]**
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations

Course Roadmap

Our next section is Windows Endpoints.

What We Cover



- Concerned about architecting better secured endpoints
 - With a goal of more fully supporting NSM, CSM, and instrumenting a SOC environment
- Overwhelmingly, the predominant endpoint found in enterprises remains Windows
- Today's material will be primarily Windows-based desktops
- Likewise, the presumption will be securing an Active Directory environment

Continuous Monitoring and Security Operations

10

What We Cover

The name of the game today is endpoint security architecture. Given the emphasis that we place on modern threats, detection, and response, it should come as no surprise that we will employ a pragmatic approach that attempts to support these emphases.

Given the ubiquity of Windows environments, our primary emphasis, where specificity is required, will be on security of Windows-based endpoints. Further, we presume that these Windows-based assets are deployed in an Active Directory infrastructure.

Endpoints – More Than Windows

- Certainly there are more OS than Windows that need some cyber defense love
- Windows still predominant OS in enterprises
- Windows also still primary target of adversaries
 - Possibly b/c it is most common enterprise OS
- Principles of defending Windows applicable to other OS
 - Though perhaps not as common or available

Continuous Monitoring and Security Operations

11

Endpoints – More Than Windows

Needless to say, there are other endpoint OSes than Windows. Shocking, I know... However, almost every organization will include a significant Windows deployment within the enterprise. For that reason, Windows systems play some role within almost every major intrusion campaign. Adversaries still emphasize Windows, perhaps because of its ubiquity.

However, just because we will emphasize some Windows-specific elements does not negate the benefit to less Windows-centric organizations. Much of the content will not be unique to Windows. Also, many of the seemingly Windows-only concepts are more widely applicable than first perceived.

Endpoints – More Than Desktops

- Wait, what about those critical servers?
- Here's a secret...desktops are much more difficult to secure than servers
 - B/C users are insane
 - B/C users are evil (not necessarily on purpose)
 - B/C they are easier to reach for adversaries
- If you can secure a desktop then you should be able to secure a single-purpose, headless, server OS

Continuous Monitoring and Security Operations

12

Endpoints – More Than Desktops

Our primary focus will be on Windows, especially Windows desktops. Wait a second: servers are more likely to be the final repository of valuable data, so why should emphasis be placed on the desktop rather than the server? The trick is that desktops are vastly more difficult to secure than servers. If, through this book's content, you are better able to secure Windows desktops, then you will necessarily have increased your facility to secure Windows servers.

The primary distinguishing feature that make desktops more challenging than servers to secure is simple: users. Active users drastically change the security posture of a system. They want to install applications, access data/resources, and also provide a more obvious conduit for adversaries to introduce their malicious content.

Endpoints: Beyond Desktops/Servers

- There are, of course, incredibly important systems that don't easily fall under the desktop or even the server category
 - Network appliances,
 - SCADA,
 - Mobile devices, etc.
- Largely the approach employed for desktops and supported by the network will be directly applicable to all other systems
 - There would be special purpose additional measures that might be warranted, but these are the exception

Continuous Monitoring and Security Operations

13

Endpoints: Beyond Desktops/Servers

Desktops, even when coupled with servers, still represent a scratched surface of what an organization must needs secure. There are also network appliances, SCADA systems, VoIP systems, mobile devices, web applications, and many more.

While there are naturally some specific differences for each individual application/device many of the underlying principles are the same as we find with Windows desktop systems.

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- **[Patching]**
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 14

Course Roadmap

Our next section is on the joy of Patching.

Patching

- B..O..R..I..N..G!!!
- Sorry, but patching is the single most important security aspect of securing enterprises
- ...and you are not nearly as good at this as you can/should be
- The overwhelming majority of compromises start with exploitation of a flaw
 - A flaw that could have been patched, but wasn't

Continuous Monitoring and Security Operations

15

Patching

We need to talk about patching. Probably one of the least exciting things we could possibly talk about, and yet, also easily one of the most important.

Though 0 day exploits seem to be becoming more common to be discovered and sold¹, luckily, the fact remains that the overwhelming majority of all exploits begin with abusing a known flaw that simply has not been patched, even though a patch was available. You are, quite frankly, almost certainly not as good at patching as you can be or need to be.

[1] <https://www.nssllabs.com/reports/known-unknowns-0> (http://cyber.gd/511_391) QR



<48 Hours

- Less than 48 hours after a patch's release it is expected that you will have it installed
- !?!?!?!?!?!?!?
- This is our target for both OS and application patches according to the Critical Security Controls
- The author regularly sees critical production systems that haven't been patched in more than 48 MONTHS (Windows NT just will not DIE!!!)
- How does your organization stack up against this target goal of <48 hours?

Continuous Monitoring and Security Operations

16

<48 Hours

If you think your organization has a fairly robust patching process, usually all it takes to reconsider this notion is to evaluate your org in comparison to the Critical Security Controls recommendation. <48 hours. That is the metric used by the CSC as the gold standard. Note that patching is considered to be so incredibly important that 2 of the 5, First Five Quick Wins, relate directly to patching.

<48 hours is the target for both Operating Systems and also for applications. This should be the target for all systems, not just public facing systems such as web servers. The adversaries shift to client-side exploitation + pivoting has made our internal systems much more reachable than many seem willing to admit.

Nation States, 0-days, and APT, Oh My!

- Certainly there are adversaries that have the ability to create (or purchase) custom exploits that are undiscovered
 - This is the stuff of 0-days
- Still don't want to make advanced adversaries jobs any easier
- And we have to deal with many less-advanced adversaries in addition to possibly advanced threats
- 0-days, unless you consider custom web application exploits, are still relatively rare
- Also, we are not suggesting that patching is all that you do, it is just a vastly important, necessary first step
 - That isn't focused on as much because it isn't terribly sexy

Continuous Monitoring and Security Operations

17

Nation States, 0-days, and APT, Oh My!

Yes, it is true that 0-day exploits are increasingly within reach of well-funded or advanced adversaries. These can either be independently developed or procured for a fee from numerous well-known organizations that offer 0day exploits as a product/service regardless of the purchasers intended use case.

Don't let the, still unlikely, potential for a 0-day exploit to distract you from rapidly trying to achieve robust patching processes that could support a <48 hour installation model.

To Test, or Not to Test

- Patch testing seems obvious and necessary, but why do we do it
 - To ensure that the cure isn't worse than the disease...
 - Because one person got a BSOD 10 years ago...
- Recall CSC goal of <48 hours, now we are going to both test and deploy less than two days after identification
- Do we really test patches, or do we just tell that to ourselves and our auditors?
 - What about all those antimalware patches, I mean updates?
- What does testing look like?
 - Most organizations simply deploy to a less critical group of production systems

Continuous Monitoring and Security Operations

18

To Test, Or Not to Test

A major concern comes up when discussing the shortened patch window suggested by CSC, patch testing. While patch testing might seem both obvious and necessary, let us consider why exactly do we actually test our patches. The basic idea, of course, is that we want to ensure that we do not inadvertently cause a negative operational impact with our patch. If you have been involved in information security long enough, you likely will recall at least one instance, perhaps more, of a Microsoft provided patch causing the dreaded BSOD (Blue Screen of Death).

Patch testing, on the surface, seems like a no-brainer, but how exactly do we achieve patch testing? Typically, patch testing simply means pushing installs to less important production systems and waiting a set period of time for notification of catastrophic failure. Barring notification, patch deployment continues. This doesn't seem to be a terribly robust process, and often only is employed for some of the easier to install patches. One question I often ask of organizations is how and whether they test updates (read: patches) of their antivirus/antimalware solution. Typically, after a few furtive glances, they indicate what almost every organization does, that no testing is employed for AV patches.

Unfortunately, we have seen a number of antimalware updates also cause operational issues or even a BSOD¹.

Given the relatively poor, and often inconsistent, process for patch testing, would it be worthwhile to consider abandoning it altogether? Personally, I am of the opinion that rather than the laughable

excuse for patch testing, an organization focus on being able to rapidly recover from any potential operational issues incurred.

[1] <https://kc.mcafee.com/corporate/index?page=content&id=KB75956> (http://cyber.gd/511_323) QR



Patch, Rinse, Repeat

- The never-ending cycle
 - Patch identification
 - Possible patch testing
 - Patch deployment
 - Patch verification
- Cycle is certainly tedious, but vastly important
- Much of an organization's security is dependent upon good patching practices
- Honestly requires dedicated staff in most organizations

Continuous Monitoring and Security Operations

20

Patch, Rinse, Repeat

The joyless patch cycle process is a never-ending soul-crushing process. The ongoing process starts with patch identification. Then it moves into possible patch testing, see previous for commentary on testing or not testing. Next up, we have patch deployment where patches are installed on the systems. The final phase involves patch verification.

Patch verification serves to ensure that the patches have been successfully installed on all systems. The basic process often simply leverages the patch management console, at least initially. However, getting a second opinion for this incredibly important aspect is warranted, and simple. By leveraging a vulnerability scanner, the organization can rapidly get a second opinion as to whether patch installation was successful and hit all systems.

Modern Patching Challenges

- How do you handle systems that you don't own, but that use your data (e.g. BYOD)?
- How do you patch systems that spend more time away from the network than on the network?
- How do you handle patching unknown applications (Google Chrome install doesn't require Admin privileges)?
- Detailed answers are beyond the scope of this section, but wanted to at least present for consideration some of the questions

Continuous Monitoring and Security Operations

21

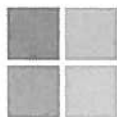
Modern Patching Challenges

Though traditional patching, especially considering third-party application patching, can be difficult enough, there are additional challenges that crop up when considering the modern enterprise landscape.

Some particularly challenging issues are found in:

- Mobile devices
- Highly portable devices
- Unknown applications

Dealing with these three patch recipients can prove fiendishly difficult. Much of the difficulty is part of a larger challenge around hardware and software inventory, which will be considered later.



Patching Microsoft

- Your organization probably has lots of Microsoft systems
- Most organizations also first really became aware of computer exploitation when faced with worm/virus outbreaks that infected Windows-based machines
 - This black eye has been oddly helpful for MS products when it comes to patching
- Organizations are typically far better at patching Microsoft than anything else that they maintain
- Unfortunately, many organizations seem to still operate as if Microsoft is the only product that needs robust patching products/processes
- Steaming cup of Oracle Java, anyone?

Continuous Monitoring and Security Operations

22

Patching Microsoft

Typically the easiest of all possible patching scenarios involves the patching of Microsoft. For many years Microsoft has provided for automatic updates targeting both consumer and enterprise installations. Additionally, Microsoft was also one of the first organizations to define a consistent patch release schedule, defined as the second Tuesday of the month.

The long history of exploitation and adversary focus on Microsoft has provided ample opportunity for Microsoft to develop strong patching practices. To help organizations achieve a more rapidly and consistently patched enterprise, Microsoft first provided Software Update Services, the precursor to Windows Server Update Services.

WSUS

- Windows Server Update Services (WSUS) is in large part responsible for why we are so good at patching Microsoft
 - And also so poor at patching other stuff
- WSUS, formerly called SUS, was released to help organizations patch the Windows OS
 - And it is FREE, FREE, FREE!!!
- It is FREE and patches what folks predominantly worry about
 - It cures what ails us, and for cheap!



Continuous Monitoring and Security Operations

23

WSUS

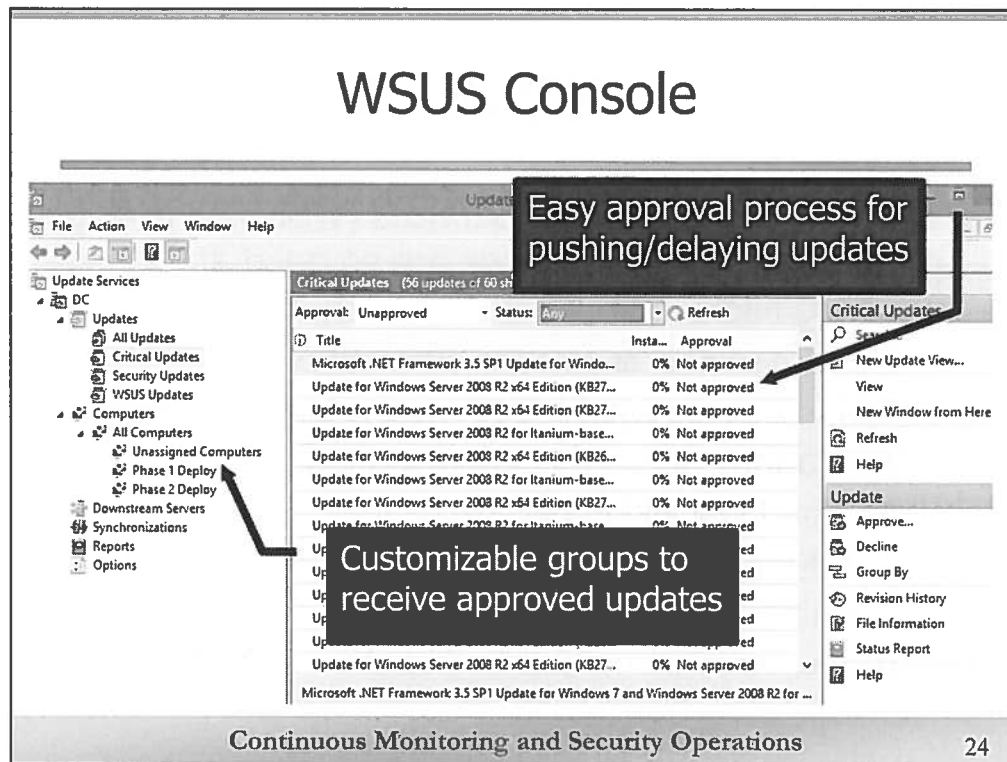
Windows Server Update Services (WSUS), which replaced the earlier Software Update Services (SUS), is Microsoft's free server that can be leveraged to serve as an organization's internal, self-managed, version of Windows Update. WSUS provides a central console for pushing updates to Windows, Office, as well as most other Microsoft products.

Note, however, that WSUS cannot easily be used to push third-party patches without additional software. Though Microsoft has exposed the ability to push third-party patches through WSUS¹, the process is unwieldy.

<http://technet.microsoft.com/en-US/wsus> (http://cyber.gd/511_31) **QR**

<http://msdn.microsoft.com/en-us/library/bb902470%28VS.85%29.aspx> (http://cyber.gd/511_370)





WSUS Console

Above we see the WSUS Console, where administrators can approve (or not) patches for deployment to managed systems. Whether systems are managed or not is controllable through group policy.

An additional capability that can prove quite useful is the ability to easily create custom groupings of managed computers. This allows for easier management of which software is deployed to which systems. This feature is commonly used to try to achieve a phased approach to deployment (which some folks seem to call patch testing).

<https://vlabs.holysystems.com/vlabs/technet?eng=VLabs&auth=none&src=microsoft.holysystems.com&altadd=true&labid=10881> (http://cyber.gd/511_33) **QR**



WSUS Good/Bad/Ugly

- WSUS has evolved to handle patching pretty much all Microsoft products
- Good:
 - Amazingly it is still FREE!
 - No additional client software is required
 - Console allows for phased/delayed patch installs
- Bad:
 - Third-party patch deployment is cumbersome without additional tools^{1,2}
 - It is good enough that folks have trouble cost justifying something else

Continuous Monitoring and Security Operations

25

WSUS Good/Bad/Ugly

The most compelling feature of WSUS is the fact that, for FREE, it can be used to provide patch management for many different Microsoft products. All of this can be done without requiring any additional client software to be installed.

The most serious detriments of using WSUS are that third-party software is not easily deployed without additional tools, most of which are commercial. Also, and this is a bit of a back-handed complement, but WSUS has been good enough for some organizations to avoid the purchase of a more feature complete offering.

[1] <http://wsuspackagepublisher.codeplex.com/> (http://cyber.gd/511_371) **QR**

[2] <http://sourceforge.net/projects/localupdatepubl/> (http://cyber.gd/511_372)



Patching Third-Party Applications

- We can hardly blame Microsoft for making WSUS good enough that we don't do so well at patching Third-party apps
- Must realize the need for patching much more than just Microsoft though
 - Oracle Java
 - Adobe Flash
 - Adobe Acrobat Reader
 - Browser Extensions
- These all need the same level of patch loving we bestow upon Microsoft

Continuous Monitoring and Security Operations

26

Patching Third-Party Applications

One of the major strikes against WSUS is that natively it does not easily allow an organization to push patches for third-party, non-Microsoft, products. When most organizations initially adopted SUS/WSUS, Microsoft was the primary concern from a vulnerability perspective.

Now organizations would be foolish to continue primarily emphasizing only Microsoft products. While certainly Microsoft products still need patching, the fact remains that Oracle Java, Adobe Acrobat Reader, Adobe Flash, and many others require significant attention in their own right.

SCCM

- Chances are good that your organization already has an SCCM (System Center Configuration Manager) deployment
- Many Microsoft shops have used SCCM to deploy one off applications packages to their systems
 - Patch Management can simply be seen as an extension to this functionality
- SCCM uses the same means as WSUS for actually pushing the packages, namely BITS



Continuous Monitoring and Security Operations

27

SCCM

Many Microsoft heavy shops, which is most organizations, tend to already have licenses for Microsoft System Center Configuration Manager (SCCM), which was formerly known as SMS. SCCM can be employed to push one-off third-party patches. SCCM has often been used by shops lacking additional patch management solutions for this purpose.

The backend patch installation technology is the same as that employed by WSUS. Pushing third-party patches with SCCM does not cost anything beyond the regular SCCM licensing costs. SCCM was not designed overtly for patch management and this becomes obvious when trying to manage patches for a large number of third-party products. While it can be effective, it does not feel like a replacement for a patch management solution.

<http://technet.microsoft.com/en-US/systemcenter> (http://cyber.gd/511_32) QR



SCUP

- Straight SCCM feels a bit unwieldy for regular third-party patch deployment
- System Center Updates Publisher (SCUP) is an application that enables more coordinated SCCM-based patch deployment
 - Offers a more robust interface
 - Supports vendor-supplied update catalogs (most-notably Adobe)
 - Allows definition of prerequisite and superseded software
- SCUP requires and depends on WSUS/SCCM install

Continuous Monitoring and Security Operations

28

SCUP

Another option exists beyond the vanilla SCCM offerings. Microsoft has a standalone extension to SCCM targeted specifically for better enabling third-party application management.

SCUP, or System Center Updates Publisher, works along side SCCM and WSUS and attempts to ease the process of applying third-party patches and updates.

One of the most compelling aspects of SCUP are the catalogs provided by some major external vendors. Most notably, Oracle Java and Adobe Reader have update catalogs available for them that can specifically plug into SCCM/SCUP to greatly ease deployment/administration.

<http://technet.microsoft.com/en-us/systemcenter/bb741049.aspx>)http://cyber.gd/511_324) **QR**



WSUS and Third-Party Apps

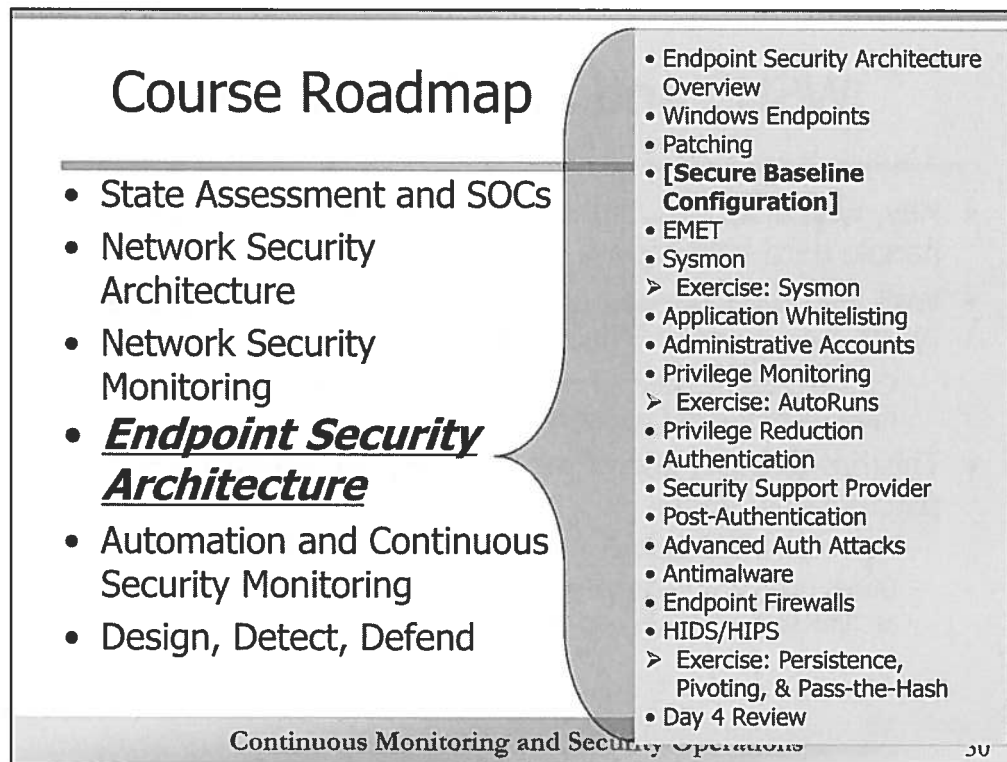
- Hey, wait a second, didn't we decide that WSUS didn't handle third-party apps...
- Well, Microsoft opened up the ability for vendors to extend WSUS to support the distribution of third-party applications
 - Can obviate the need for SCUP and SCCM (though you probably have that for other reasons)
- This has become one of the most popular approaches to patch management
 - No need for yet another standalone agent to manage
 - Builds upon a well-tested and scalable software deployment architecture of WSUS and SCCM

WSUS and Third-Party Apps

Now this might seem a bit odd, but we can actually use vanilla WSUS to deploy third-party application updates. Seems a bit odd given that a few pages back we discussed the fact that WSUS cannot accommodate third-party, and that SCCM/SCUP must be used, if staying purely Microsoft, to achieve this.

Nothing has changed on the Microsoft third-party application publishing front, they still will not provide this functionality directly in WSUS without SCCM/SCUP. However, Microsoft did open up the ability for vendors to leverage WSUS and BITS for third-party patch deployment.

Effectively, what this means is that other vendors can leverage your existing WSUS deployment for management, and will not even necessarily require an additional agent on the client-side, which is a significant improvement, from my vantage point.



Course Roadmap

Our next section is Security Baseline Configuration.

{CSC} Baseline Configuration

- What is better than patching an application?
 - Not having the application in the first place
- All systems/applications are vulnerable
 - Whether you know the vulnerabilities or not is different concern
- We will inevitably overlook or have issues with particular patch installations
- We will have endpoints that are routinely beyond the reach of our robust network security architecture
- The best security in those cases is having a well-vetted hardened baseline configuration

Continuous Monitoring and Security Operations

31

Baseline Configuration

Every system and application has vulnerabilities. Now, at times, we might not be aware of any vulnerabilities that are lacking a patch, but the fact remains that they still exist. In time, adversaries, researchers, the vendors, or someone else entirely will discover a flaw. After details are reviewed, a patch could then be created and made available.

This brings us back to our previous section and discussion on the joys of patching. The endless cycle repeats again.

However, what if we were able to identify software that was not needed by the organization? Then we could remove the software, and thus obviate the need to patch that software. Further, what if the flawed component of the application was functionality that had been explicitly disabled in our environment. Even without a patch, the risk might well have been successfully mitigated, even without having first patched the flaw.

The baseline configuration seeks to determine the required and necessary components of systems and software, and no more.

Building a Baseline Config

- Several goals of the baseline configuration
 - Determine a reasonably secure starting point for systems' configurations
 - Establish a consistent configuration across majority of systems
 - Reduce time to recover a deployed system
- The impact of a baseline config is significant and much time and care should be taken during the building of the config

Continuous Monitoring and Security Operations

32

Building a Baseline Config

Though a security baseline configuration sounds conceptually simple, actually finding the balance between the best security and the easiest usability is consistently a challenge.

The overarching goals are:

Identify the necessary components that comprise a baseline configuration of a particular system, application, technique

Establish a consistent configuration deployed throughout the organization

Reduce the business impact and time to recovery of a fielded system

Much like patching, baseline configuration is typically not one of the most exciting projects a security professional can be tasked with. However, the importance of solid practices on this front cannot be overstated.

How NOT to Build a Config

- Start from scratch and figure out all of the needed settings through trial and error
- Simply reuse, in its entirety, a vendor or other organization's provided config
- Deploy the most hardened possible configuration known to humans
- Exert tremendous effort once and think that you have got this config management thing done

Continuous Monitoring and Security Operations

33

How NOT to Build a Config

There are some key recipes for failure on the development of a baseline config.

One of the first way that an organization can quickly have an abandoned project is to try to build the config of a modern system or application from scratch. An application or system of any considerable size proves fiendishly difficult to understand at a level sufficient to decide for yourself the best configs.

The other end of the DIY spectrum involves organizations that simply try to use someone else's opinion entirely as to the proper configuration.

Another fail that some junior security professionals stumble upon is to err on the side of the most hardcore security conscious configuration possible, that can clearly not operate in any normal organization.

The final common failure is to simply consider this to be a one time process. In truth, this process does require much more upfront skill and labor, but it also necessarily requires ongoing care and feeding to ensure continued relevance and applicability.

Center for Internet Security

- Leveraging an established third-party configuration as a starting point is a good choice
- The Center for Internet Security's benchmarks have long been a trusted source for good security baseline configurations
- The benchmarks are developed via consensus from an working group from industry
- The benchmarks are provided free-of-charge
- Importantly, there is significant documentation and guidance explaining the various settings and their potential implications

Continuous Monitoring and Security Operations

34

Center for Internet Security

Easily the most well-known and highly regarded starting point for baseline configurations come from the Center for Internet Security (CIS). CIS provides what they refer to as benchmarks for myriad software from full operating systems to some specialized but popular applications as well as hardware appliances. One of the most compelling features of CIS Benchmarks is that they are so vast in their coverage including iOS 7, Internet Explorer, Microsoft Office, Apple OSX, Windows 8, HP-UX, FreeBSD, VMware ESX, Microsoft Exchange, and many more.

These benchmarks provide guidance on the secure configuration of the software/hardware being referenced. Beyond the scope of the benchmarks, another very significant feature of CIS is that they are developed by consensus of experts that are not all from the vendor. CIS is a not-for-profit and the benchmarks themselves are provided free-of-charge to the community.

The documentation provided in the benchmarks, for free, is extremely good and supplies guidance on why you would or would not be advised to adhere to their recommended settings.

<https://benchmarks.cisecurity.org/> (http://cyber.gd/511_325) QR



CIS Benchmarks

- To better illustrated the vast coverage provided by CIS, currently available CIS Benchmarks are provided in the notes



Continuous Monitoring and Security Operations

35

CIS Benchmarks

Apache HTTP & Tomcat Benchmarks

Apple iOS Benchmarks

Apple OSX Benchmarks

Apple Safari Benchmarks

CentOS Linux Benchmarks

CheckPoint Firewall Benchmarks

Cisco Device Benchmarks

Consensus Security Metrics

Debian Linux Benchmarks

FreeBSD Benchmarks

FreeRadius Benchmarks

Google Android Benchmarks

HP-UX Benchmarks

IBM AIX Benchmarks

IBM DB2 Benchmarks

ISC BIND Benchmarks

Juniper Device Benchmarks

Kerberos Benchmarks

LDAP Benchmarks

Microsoft Exchange Server Benchmarks

Microsoft IIS Benchmarks

Microsoft Internet Explorer Benchmarks

Microsoft MS SQL Server Benchmarks

Microsoft Office Benchmarks

Microsoft SharePoint Server Benchmarks

Microsoft Windows 7 Benchmarks

Microsoft Windows 8 Benchmarks

Microsoft Windows NT Benchmarks

Microsoft Windows Server 2000 Benchmarks

Microsoft Windows Server 2003 Benchmarks

Microsoft Windows Server 2008 Benchmarks

Microsoft Windows Server 2012 Benchmarks

Microsoft Windows XP Benchmarks
Mozilla Firefox Benchmarks
Multi Function Print Devices Benchmark
MySQL Database Server Benchmarks
Opera Benchmarks
Oracle Database Server Benchmarks
Oracle Solaris Benchmarks
Red Hat Linux Benchmarks
Router Assessment Tool
Slackware Linux Benchmarks
SuSE Linux Benchmarks
Sybase ASE Benchmarks
Ubuntu Linux Benchmarks
Virtualization Benchmarks
VMware Benchmarks
Xen Benchmarks¹

[1] <https://benchmarks.cisecurity.org/downloads/multiform/> (http://cyber.gd/511_326) **QR**



Vendor Guides

- The CIS Benchmarks provide tremendous insight from a vendor neutral vantage point
- Where available, vendor guides should also be consulted though
- Be mindful that the quality of vendor guides can vary rather drastically depending upon the vendor
 - At times the quality of security guidance can even differ across products from one vendor

Continuous Monitoring and Security Operations

37

Vendor Guides

An increasingly common source of security guidance comes from the vendors themselves. Though CIS provides ample coverage from a vendor neutral standpoint, there is necessarily some lag between the release of the hardware/software and the development or update of the benchmark guide from CIS.

Many vendors now provide their own guidance on securing their products. From one standpoint, who better is positioned to provide expert opinion on the most secure configuration of a product than the vendor? However, some also speculate that the vendor might be less inclined to provide any guidance that limits functionality that serves the vendor's interest without regard to the security implications.

More important than any notion of vendor disincentives for security configurations is simply the quality of guidance. There is naturally a rather significant difference in quality from one vendor security configuration guide to the next. Care must be taken.

Microsoft Security Compliance Manager (SCM)

- Microsoft's approach to distributing guidance on security benchmarks
- Grown into a much more substantial offering than simply a collection of security guides
- SCM also includes tools and scripts to facilitate implementation of the suggested guidance

Continuous Monitoring and Security Operations

38

Microsoft Security Compliance Manager (SCM)

Easily the most well-known vendor security guides are produced in Redmond, Washington. Microsoft has a fairly substantial history of providing security configuration guidance for many of their products. Historically, this has been simply by providing some basic security templates and a guide that could be used within the larger Microsoft, and Active Directory, ecosystem to ease configuration of security-relevant settings and features.

Now, with Microsoft's Security Compliance Manager, the folks from Redmond have created a much more robust offering. Not only are updated security guides provided, but there are also tools for importing existing system configuration and comparing them against the guidance in the security guides. Additionally, tools are provided to establish a baseline that can be deployed via domain GPO and also to standalone systems not part of the domain.

<http://technet.microsoft.com/en-us/library/cc677002.aspx> (http://cyber.gd/511_34) QR



Beyond Vendors and CIS

- Governments throughout the world also sometimes weigh in on security configuration guidance
- The United States government has several different organizations that have provided guidance in the past
 - The NSA also produces Security Configuration Guides for many different vendors' products as well (**QR**)
 - NSA's IAD also provides fact sheets and other guidance at the same location (e.g. Reducing the Effectiveness of Pass-the-Hash, Spotting the Adversary with Event Log Monitoring, etc.)
 - DISA STIGs (Security Technical Implementation Guides) represent the most common security configuration guidance produced by the US government

Continuous Monitoring and Security Operations

39

Beyond Vendors and CIS

Traditional vendor security guides and CIS are still not the extent of offerings on security configuration guidance. Other third-parties also provide their own take on security configuration guidance. Two of the most well-known are available free from the US government.

The first come from the NSA and are not exclusively guidance about products. The NSA Security Configuration Guides also include some fairly compelling fact sheets or point guidance. An example of guidance includes, "Reducing the Effectiveness of Pass-the-Hash"¹ or "Spotting the Adversary with Event Log Monitoring"². Many security professionals seem unaware of these offerings and largely only know of the security templates in passing.

DISA (Defense Information Systems Agency) also provides STIGs (Security Technical Implementation Guides), which are the most commonly used configuration guides in the US government. Note that some of the guides are FOUO (For Official Use Only) and would require a DoD supplied PKI cert to access. However, most of the guidance is Unclassified and can simply be downloaded directly. STIGs are also intended to be assessed systematically and so provide the configuration files in a format that is parsable with SCAP-capable scanners or scripts.

[1] http://www.nsa.gov/ia/_files/app/Reducing_the_Effectiveness_of_Pass-the-Hash.pdf
(http://cyber.gd/511_327)

[2]

http://www.nsa.gov/ia/_files/app/Spotting_the_Adversary_with_Windows_Event_Log_Monitoring.pdf (http://cyber.gd/511_328)

http://www.nsa.gov/ia/mitigation_guidance/security_configuration_guides/index.shtml (http://cyber.gd/511_317) **QR**

<http://iase.disa.mil/stigs/> (http://cyber.gd/511_318)



Configuration Change Monitoring

- Starting with a strong security configuration is meaningless if changes are not controlled over time
- You certainly have an approval process, perhaps even a Change Control Board, but amazingly unauthorized changes still occur
 - Changes could be malware
 - Or an overzealous admin
 - Or often the will of management
- Controlling and monitoring for security relevant changes is vital

Continuous Monitoring and Security Operations

41

Configuration Change Monitoring

Perhaps even more important than establishing the initial security baseline configuration is systematically managing the changes to the baseline. Every new application, configuration change, or update could impact the effective security posture. Most organizations fail rather miserably at truly managing the changes.

These failings exist in spite of the existence, at least in larger organizations) of a Change Control Board (CCB) that is intended to be knowledgeable of, and moreover provide guidance on, these changes. Given the speed with which systems' configurations can change technical controls are needed to complement or mitigate the risk of changes flying under the radar of the CCB.

Baseline Monitoring

- An extremely important tool for strong cyber defense is monitoring our systems for configuration changes
 - Not simply talking about file integrity monitoring
 - Also not talking about from a change control or audit perspective
- Configuration monitoring for cyber defense
 - Watching key aspects of the system configuration over time and analyzing those changes
 - Looking for security relevant changes or seeing what changes have occurred after a compromise
- This is a significant chunk of what we will be doing when we move into the Continuous Security Monitoring portion of the course

Continuous Monitoring and Security Operations

42

Baseline Monitoring

The technical control over this process involves robust and proactive monitoring for key security relevant changes. The goal is not to monitor for auditing's sake, which is often the primary focus of the Change Control Board. Rather the goal is a practical security goal of ensuring that the organization is operating under the correct assumptions about their security posture.

Consider simply have a daily (weekly or even monthly would likely be a vast improvement) report for each system that highlights key aspects of the system users, services, ports, installed applications, binaries, and others. First, let's keep it easy and simply archive all of this information for later review. Given a suspected compromise, simply review the output of the reports and diff them over time to get a sense of what has changed, and when it could have changed. This is a great boon to both incident response and post-mortem forensics.

Though Configuration Monitoring can be a significant aid when performing Incident Response or even post-mortem forensics, instrumented properly these reports can provide for rapid detection. Imagine scripts continuously monitoring for these changes over time and alerting on significant ones.

We will be doing much of this during the Continuous Monitoring portion of the course.

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- **[EMET]**
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 43

Course Roadmap

Our next section is Host and Application Discovery.

EMET

- Microsoft's EMET (Enhanced Mitigation Experience Toolkit) is a tool that hardens Windows operating systems against a series of common exploit tactics
 - Free download from Microsoft
- Can be used to harden any version of Windows from XP and 2003 through Windows 10 and Server 2012
- Especially helpful for helping protect legacy operating systems such as Windows XP (end of life) and Windows Server 2003 (end of life)
- *EMET is not a magic bullet. It is designed with two goals: to raise the cost of exploit development, and to reduce or eliminate the efficacy of existing pre-written shellcode.*¹

Continuous Monitoring and Security Operations

44

EMET

EMET may protect any reasonably recent Microsoft operating system, from XP on up. It is especially helpful for legacy operating systems such as XP and Server 2003. Enterprises should obviously upgrade these systems, but the reality is XP is still very common in the enterprise. EMET adds some protection to these weak systems.

EMET 5.2 can be used for Windows Vista through 8.1 (older versions of EMET may be used to protect XP, etc.). As of this writing, EMET5.5 beta (installed in the Sec-511-Windows-10 VM) can be used to protect Windows 10.

Microsoft describes EMET:

*The Enhanced Mitigation Experience Toolkit (EMET) is a utility that helps prevent vulnerabilities in software from being successfully exploited. EMET achieves this goal by using security mitigation technologies. These technologies function as special protections and obstacles that an exploit author must defeat to exploit software vulnerabilities. These security mitigation technologies do not guarantee that vulnerabilities cannot be exploited. However, they work to make exploitation as difficult as possible to perform.*²

[1] <http://security.stackexchange.com/questions/68691/what-does-emet-do-for-windows-8-1> (http://cyber.gd/511_383)

[2] <http://support.microsoft.com/en-us/kb/2458544> (http://cyber.gd/511_384)

EMET Features

- Big area of focus: back-port newer security controls (such as DEP, ASLR and ROP mitigation) to older operating systems that lack these features natively
- Includes a large list of controls:
 - Attack Surface Reduction (ASR) Mitigation
 - Export Address Table Filtering (EAF+) Security Mitigation
 - Data Execution Prevention (DEP) Security Mitigation
 - Structured Execution Handling Overwrite Protection (SEHOP) Security Mitigation
 - NullPage Security Mitigation
 - Heapspray Allocation Security Mitigation
 - Export Address Table Filtering (EAF) Security Mitigation
 - Mandatory Address Space Layout Randomization (ASLR) Security Mitigation
 - Bottom Up ASLR Security Mitigation
 - Load Library Check – Return Oriented Programming (ROP) Security Mitigation
 - Memory Protection Check – Return Oriented Programming (ROP) Security Mitigation
 - Caller Checks – Return Oriented Programming (ROP) Security Mitigation
 - Simulate Execution Flow – Return Oriented Programming (ROP) Security Mitigation*
 - Stack Pivot – Return Oriented Programming (ROP) Security Mitigation¹

Continuous Monitoring and Security Operations

45

EMET Features

This list is extensive, and includes protection against cutting-edge techniques such as ROP (Return Oriented Programming).

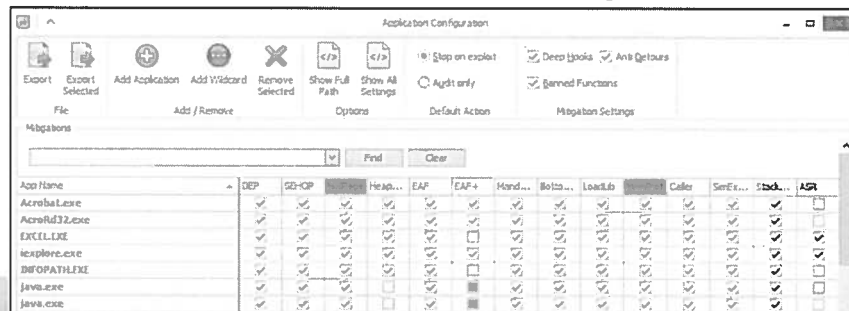
EMET 5.2 was released in march 2015, details are available here:
<http://blogs.technet.com/b/srd/archive/2015/03/12/emet-5-2-is-available.aspx>
(http://cyber.gd/511_386)

The initial release of EMET 5.2 caused issues with Windows 8.1 and Internet Explorer. Microsoft fixed the problem, and recommends that anyone who downloaded EMET 5.2 on or before March 16th 2015 download it again.

[1] <https://technet.microsoft.com/en-us/security/jj653751?f=255&MSPPErr=-2147217396>
(http://cyber.gd/511_385)

EMET Configuration GUI

- Any application may be protected by EMET
 - Many are automatically covered by default
- Supports audit mode for testing applications
 - Testing is critical when adding new applications
 - Logs via Windows application event logs



46

EMET Configuration GUI

EMET is installed in the Sec511 Windows 10 virtual machine. You can load the EMET configuration GUI by searching for EMET, and launching "EMET GUI."

A number of applications are automatically protected, including third-party applications such as Java and Adobe Acrobat. You may add others to the list by clicking "Add application."

Bryan Krebs posted some good advice on adding applications to EMET:

While you're at it, add the rest of your more commonly used, Internet-facing apps. But go slow with it, and avoid the temptation to make system-wide changes. Changing system defaults across the board – such as changing ASLR and DEP settings using the "configure system" tab – may cause stability and bootup problems.¹

[1] <http://krebsonsecurity.com/2013/06/windows-security-101-emet-4-0/> (http://cyber.gd/511_387)

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- **[Sysmon]**
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 4 /

Course Roadmap

Our next section is Host and Application Discovery.

Application Monitoring

- We will discuss application whitelisting in the next section
 - This is the best endpoint control you are (probably) not using
- For sites that haven't deployed whitelisting: monitoring application use on critical servers is paramount
- Many malware attacks involve dropping binaries onto systems and running them
 - Mimikatz is a notable example, discussed shortly

Application Monitoring

We will discuss application whitelisting in the next section. This is the best endpoint control you are (probably) not using.

For sites that haven't deployed whitelisting: monitoring application use on critical servers is paramount.

Many malware attacks involve dropping binaries onto systems and running them. Mimikatz is a notable example, discussed shortly.

Microsoft Sysinternals Sysmon

- Sysinternals Sysmon is a great free tool that monitors application use (and more)

System Monitor (Sysmon) is a Windows system service and device driver that, once installed on a system, remains resident across system reboots to monitor and log system activity to the Windows event log. It provides detailed information about process creations, network connections, and changes to file creation time. By collecting the events it generates using Windows Event Collection or SIEM agents and subsequently analyzing them, you can identify malicious or anomalous activity and understand how intruders and malware operate on your network.¹

Continuous Monitoring and Security Operations

49

Microsoft Sysinternals Sysmon

Sysmon 2.0 was released in January 2015:

This major update to Sysmon, a service that records process activity to the Windows event log for use by incident detection and forensic analysis, includes driver load and image load events with signature information, configurable hashing algorithm reporting, flexible filters for including and excluding events, and support for supplying configuration via a configuration file instead of the command line.²

Sysmon 3.0 was released in April 2015 (version 3.1 followed in Jul 2015):

This release of Sysmon, an advanced background monitor that records process-related activity to the event log for use in intrusion detection and forensics, adds the process name to process terminate events, reports remote thread creation events, and improves the simplicity and flexibility of filter settings.³

[1] <https://technet.microsoft.com/en-us/subscriptions/downloads/dn798348.aspx>
(http://cyber.gd/511_381)

[2] <https://technet.microsoft.com/en-us/sysinternals/bb545021.aspx> (http://cyber.gd/511_390)

[3] <http://blogs.technet.com/b/sysinternals/archive/2015/04/21/update-sysmon-v3-0-autornus-v13-3-regjump-v1-1-process-monitor-v3-11.aspx> (http://cyber.gd/511_487)

Sysmon Advantages

- It's free!
 - Also downloaded from Microsoft.com, which may ease introduction into locked-down environments
- Integrates cleanly into most SIEM or Windows Event Collection environments
- Can log the full command line of every launched process
- Can automatically generate checksums of all (or selected) binaries that run on a system
 - Allows submission to services such as VirusTotal
 - Or a belt-and-suspenders detective whitelisting process...
- Includes many more features, including driver loads, image (including DLL) loads, network traffic, etc.

Continuous Monitoring and Security Operations

50

Sysmon Advantages

The following events may be logged via Sysmon:

Id	Tag	Event
1	ProcessCreate	Process Create
2	FileCreateTime	File creation time changed
3	NetworkConnect	Network connection detected
5	ProcessTerminate	Process terminated
6	DriverLoad	Driver loaded
7	ImageLoad	Image loaded
8	CreateRemoteThread	CreateRemoteThread detected

Note that ImageLoad and NetworkConnect logging are disabled by default, the rest are enabled by default

[1] <https://technet.microsoft.com/en-us/subscriptions/downloads/dn798348.aspx>
(http://cyber.gd/511_381)

Sysmon Syntax

- **sysmon -i**
 - Install sysmon service and driver
- **sysmon -c**
 - Print current configuration
- **sysmon -c config.xml**
 - Load configuration from XML file
- **sysmon -l**
 - Log modules (may impact system performance due to high number of events)
- **sysmon -n**
 - Log network connections
- **sysmon -? config**
 - List detailed configuration help
- Full syntax described in notes

Continuous Monitoring and Security Operations

51

Sysmon Syntax

Install: C:\Windows\sysinternals\Sysmon.exe -i <configfile> [-h <[sha1|md5|sha256|imphash|*],...>] [-n (<process,...>)] [-l (<process,...>)]

Configure: C:\Windows\sysinternals\Sysmon.exe -c <configfile> [--|[-h <[sha1|md5|sha256|imphash|*],...>] [-n (<process,...>)] [-l (<process,...>)]]

Uninstall: C:\Windows\sysinternals\Sysmon.exe -u

-c Update configuration of an installed Sysmon driver or dump the current configuration if no other argument is provided. Optionally take a configuration file.

-h Specify the hash algorithms used for image identification (default is SHA1). It supports multiple algorithms at the same time.

Configuration entry: Hashing.

-i Install service and driver. Optionally take a config file.

-l Log loading of modules. Optionally take a list of processes to track. Configuration entry: ImageLoading.

-n Log network connections. Optionally take a list of processes to track. Configuration entry: Network.

-u Uninstall service and driver.¹

[1] <https://technet.microsoft.com/en-us/subscriptions/downloads/dn798348.aspx> (http://cyber.gd/511_381)

Example Sysmon XML Configuration

```
<Sysmon schemaversion="2.0">
  <!-- Capture all hashes -->
  <HashAlgorithms>*</HashAlgorithms>
  <EventFiltering>
    <!-- Log all drivers except if the signature -->
    <!-- contains Microsoft or Windows -->
    <DriverLoad onmatch="exclude">
      <Signature condition="contains">microsoft</Signature>
      <Signature condition="contains">windows</Signature>
    </DriverLoad>
    <!-- Do not log process termination -->
    <ProcessTerminate onmatch="include" />
    <!-- Log network connection if the destination port equal 443 -->
    <NetworkConnect onmatch="include">
      <DestinationPort>443</DestinationPort>
    </NetworkConnect>
  </EventFiltering>
</Sysmon>
```

1

Sample Sysmon XML Configuration

You may view this file locally: it's in \labs\sysmon-example-config.txt on your Windows 10 VM.

Supported hash types are: SHA1 (default), MD5, SHA256 or IMPHASH.

The above syntax is from sysmon itself, you may see this (and more) by opening a PowerShell window and typing:

```
PS C:\> sysmon -? config
```

[1] <https://technet.microsoft.com/en-us/sysinternals/dn798348> (http://cyber.gd/511_381)

Sysmon 3.1

- Sysmon 3.0 was released in April 2015, followed by Sysmon 3.1 in July 2015
 - Version 3 "adds the process name to process terminate events, reports remote thread creation events, and improves the simplicity and flexibility of filter settings."¹
- Versions 3+ support a new/simpler schema
 - Current XML configurations list `<Sysmon schemaversion="2.0">`
- The old syntax had a learning curve and was counter-intuitive
 - `<Sysmon schemaversion="1.0">`
 - Had two sections: configuration and rules
 - See notes for details

Sysmon 3.1

The old (pre-version 3) Sysmon had a more convoluted syntax. Note the following does not apply to Sysmon versions 3+, it is listed here in case you have systems running older versions of Sysmon.

There were 2 basic sections: Configuration and Rules

Configuration section default setting is to **include**

- NetworkConnect and ImageLoad are disabled by default and must be enabled in the Configuration section

Rule section default setting is to **exclude**

- If you set "default=include", the following rules in the same block exclude

[1] <https://technet.microsoft.com/en-us/sysinternals/dn798348> (http://cyber.gd/511_381)

Sysmon Event Filtering

- The EventFiltering Section allows inclusion or exclusion of events
- The following event filters enables logging of drivers, but **excludes** logging drivers loaded with "Microsoft" or "Windows" in the signature:

```
<DriverLoad onmatch="exclude">  
  <Signature condition="contains">microsoft</Signature>  
  <Signature condition="contains">windows</Signature>  
</DriverLoad>
```

- The following event filter **includes** traffic sent to port 443

```
<NetworkConnect onmatch="include">  
  <DestinationPort>443</DestinationPort>  
</NetworkConnect> </EventFiltering>
```

Continuous Monitoring and Security Operations

54

Sysmon Event Filtering

The EventFiltering Section allows inclusion or exclusion of events.

The following event filters exclude drivers with "Microsoft" or "Windows" in the signature:

```
<DriverLoad onmatch="exclude">  
  <Signature condition="contains">microsoft</Signature>  
  <Signature condition="contains">windows</Signature>  
</DriverLoad>
```

The following event filter includes traffic sent to port 443

```
<NetworkConnect onmatch="include">  
  <DestinationPort>443</DestinationPort>  
</NetworkConnect> </EventFiltering>
```

Sysmon Event Filtering II

- *If the value is 'include', it means only matched events are included. If it is set to 'exclude', the event will be included except if a rule match.¹*
- This means an "include" filter with no matches will disable filtering:

```
<ProcessTerminate onmatch="include" />
```

 - Will only log matches, and there are none
 - See notes for details
- The reverse is true for exclude filters with no matches
 - Will log everything (nothing is excluded)

Continuous Monitoring and Security Operations

55

Sysmon Event Filtering II

As discussed on the previous slide, this will log traffic to port 443 only:

```
<NetworkConnect onmatch="include">  
  <DestinationPort>443</DestinationPort>  
</NetworkConnect> </EventFiltering>
```

Include means log if there is a match.

That means an include with no matches disables logging:

```
<NetworkConnect onmatch="include" />
```

Note the "/" before the closing ">". That opens and closes the NetworkConnect filter with no matches listed. Since nothing is matched, nothing is logged.

[1] <https://technet.microsoft.com/en-us/sysinternals/dn798348> (http://cyber.gd/511_381)

Detecting Unusual and Unsigned Drivers and Images with Sysmon

- Note the two sysmon event logs on the right
- One is signed (by Microsoft)
- One isn't!

```
Administrator: C:\Users\Public\Desktop\powershell.exe
TimeCreated : 10/8/2015 3:51:39 PM
ProviderName : Microsoft-Windows-Sysmon
Id : 7
Message : Image loaded:
UtcTime: 2015-10-08 19:51:39.610
ProcessGuid: {90E22FD2-C94B-5616-0000-001003E5D742}
ProcessId: 1488
Image: C:\Windows\System32\Taskmgr.exe
ImageLoaded: C:\Windows\System32\VEEventDispatcher.dll
Hashes: SHA1-A0BAEAA01483641EAF4EAD7EAF4A08519BE4AC05
Signed: true
Signature: Microsoft Windows
```

```
Administrator: C:\Users\Public\Desktop\powershell.exe
TimeCreated : 10/8/2015 3:27:05 PM
ProviderName : Microsoft-Windows-Sysmon
Id : 7
Message : Image loaded:
UtcTime: 2015-10-08 19:27:05.340
ProcessGuid: {90E22FD2-C389-5616-0000-001009A7183E}
ProcessId: 4480
Image: C:\Users\student\AppData\Local\Temp\mimikatz.exe
ImageLoaded: C:\Users\student\AppData\Local\Temp\mimikatz.exe
Hashes: SHA1-498838860604554538862F813B160593E01F301A
Signed: false
Signature:
```

Continuous Monitoring and Security Operations

56

Detecting Unusual and Unsigned Drivers and Images with Sysmon

Sysmon can log loaded images (.EXE and .DLL) and loaded drivers (.SYS). Images and drivers will be signed in most cases.

Note the two images show above. One shows a legitimate DLL loaded by Taskmgr.exe, which is signed by "Microsoft Windows". The other is Mimikatz, which is unsigned.

We will discuss Mimikatz later in 511.4, and will use Sysmon hands-on in the next lab.

Belt-and-Suspenders Detective Whitelisting Process

- Centralize Sysmon event logs via your SIEM or event log collector (more on this in 511.5)
- Collect SHA1 hashes (and others if desired) of every process launched on critical systems
 - SHA1 is supported by both VirusTotal and the National Software Reference Library (described in the next section)
- Whitelist (ignore) known good binaries
- Alert/investigate unknown binaries
- Whitelisting is superior, but this is a great middle step
 - And the price is right!

Belt-and-Suspenders Detective Whitelisting Process

We will discuss sources of hashes for known good binaries in the upcoming application whitelisting section. One great (and free) source that we will discuss is the National Software Reference Library (NSRL):

The NSRL RDS contains metadata on computer files which can be used to uniquely identify the files and their provenance. For each file in the NSRL collection, the following data are published:

- *Cryptographic hash values (MD5 and SHA-1) of the file's content. These uniquely identify the file even if, for example, it has been renamed.*
- *Data about the file's origin, including the software package(s) containing the file and the manufacturer of the package.*
- *Other data about the file, including its original name and size.¹*

[1] <http://www.nsrl.nist.gov/new.html> (http://cyber.gd/511_389)

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - **[Exercise: Sysmon]**
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 36

Course Roadmap

Our next section is Host and Application Discovery.



SEC511 Workbook Sysmon

Exercise 1: Sysmon

Continuous Monitoring and Security Operations

59

SEC511 Workbook Sysmon

Please go to the 511 Exercise Workbook, section 511.4-1.

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- **[Application Whitelisting]**
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 00

Course Roadmap

Our next section is Host and Application Discovery.

{CSC} CSC 2-1: Inventory of Authorized and Unauthorized Software

- *Quick wins (First Five - #1): Deploy application whitelisting technology that allows systems to run software only if it is included on the whitelist and prevents execution of all other software on the system.¹*

Continuous Monitoring and Security Operations

61

CSC 2-1: Inventory of Authorized and Unauthorized Software

CSC 2-1 continues:

The whitelist may be very extensive (as is available from commercial whitelist vendors), so that users are not inconvenienced when using common software. Or, for some special-purpose systems (which require only a small number of programs to achieve their needed business functionality), the whitelist may be quite narrow. When protecting systems with customized software that may be seen as difficult to whitelist, use item 8 below (isolating the custom software in a virtual operating system that does not retain infections).²

[1] <http://www.sans.org/critical-security-controls/control/2> (http://cyber.gd/511_510)

[2] Ibid.

{CSC} Application Whitelisting

- One element of the previous section focused on software inventory
 - This provided a significant potential security boon
- If we know what software has been confirmed to be authorized we can look for deviations
 - The list of confirmed authorized or known-good represents our **whitelist**
- Anything beyond the known-good list, at the very least, requires exception handling
 - Hopefully, malware will not make it as an approved exception

Continuous Monitoring and Security Operations

62

Application Whitelisting

Building upon our previous software inventory can result in tremendous security value. At the end of the software inventory there was an implied review of the inventoried software to determine whether it was authorized, and moreover, necessary.

Developing a solid, vetted, inventory of software is necessarily a time-consuming process, but also one that often results in the discovery and subsequent removal of malicious, suspicious, or simply even unnecessary software.

Conceptually, this serves as the underlying basis for our application whitelist. We want to allow a list of known-good software, that has been vetted and deemed approved. Anything beyond that list should be blocked, or, at the very least, considered suspicious until handled.

Application (not file) Whitelist

- To be clear, this quick win security control is not concerned with regular-old files
 - The whitelist doesn't care whether that critical spreadsheet has changed (File Integrity Monitoring)
- In fact, application whitelisting doesn't even care if a new malware binary is dropped into System32
 - Only becomes relevant to application whitelisting once that binary tries to run
- The focus is on executables, applications, binaries once they attempt execution
- Those files that execute code are in-scope

Continuous Monitoring and Security Operations

63

Application (not file) Whitelist

To be certain that we all appreciate what the application whitelist can, and also cannot afford us. The app whitelist does not provide direct benefits regarding the confidentiality or integrity of data. However, it does provide substantial indirect benefits on these fronts.

Even more surprising to some is that application whitelisting typically does not even help with malicious executables being written to a compromised system. Sounds odd, but the overt point of app whitelisting is to prevent someone from successfully executing that binary and does not deal directly with the placement of said malicious binary on the system in the first place.

The Whitelist

- We need to build the whitelist of known-good executables
- Once we have the list though, how do we determine if the file attempting execution is actually on the list
 - Abe Froman, Sausage King of Chicago, issue
- What happens if malware is named lsass.exe or svchost.exe, should it magically become trusted
 - For some poor configurations the answer is yes

Continuous Monitoring and Security Operations

64

The Whitelist

At the highest level the whitelisting process involves building the list of known and vetted applications, and then subsequently monitoring to see if a binary attempting to execute matches one on the list.

Conceptually simple, the devil is in the details. Imagine you have put in the time and effort to build and verify the whitelist includes nothing more or less than exactly what is needed. Now, along comes malware trying to execute. How do you actually determine if the malware is or is not on the list?

What happens if the malware in question is named lsass.exe? Just because the name matches one on the list should it run? I refer to this as the Abe Froman, Sausage King of Chicago, issue, which is a reference to Ferris Bueller's Day Off¹.

Poorly configured whitelists could actually allow malware to execute if there is even a simple name match. How we actually determine whether an executable is a match for one on the WL or not is actually a fairly significant issues in application whitelisting.

[1] <http://www.urbandictionary.com/define.php?term=Abe%20Froman>
(http://cyber.gd/511_331) QR



Whitelist Integrity

- Depending upon the software, whitelist integrity checking can be performed using various approaches
 - Filename
 - Full path + Filename
 - Publisher
 - MD5 Hash
 - SHA256 Hash
 - Digital Signature
- Choose wisely here; this has serious implications

Easier to Admin ↑ **Easier to Bypass**

Continuous Monitoring and Security Operations

65

Whitelist Integrity

The point at which software determines whether a file matches one on the whitelist or not is critical. Configured poorly, malware could possibly bypass this control with relative ease.

However, if configured to the more hardcore end of the spectrum, then the administrative burden of the maintaining the whitelist itself can cause issues.

Below is a quick list of items that can be compared to determine whether there is a whitelist match or not. The top represents the most obvious, easiest to administer, but unfortunately also easiest to bypass approaches. The bottom of the spectrum represents the highest degree of administrative overhead, but also the highest level of integrity.

- Filename
- Full path + Filename
- Publisher
- MD5 Hash
- SHA256 Hash
- Digital Signature

Typical Flow of Executables

- Once the list has been created, it will require administration and ongoing maintenance
- Need to allow for patching (of course)
 - But still need the security benefit of blocking unknown/untrusted
- One key to this is understanding the typical innocuous flow of executables in your environment
- And also the path-of-least-resistance for introduction of malicious executables

Typical Flow of Executables

One area that can prove helpful is addressing the way in which systems can, and also how they should receive executables. Unfortunately, our whitelist is not and cannot be static. Binaries will be updated over time.

We need to allow for patching/updating of the executables, and associated whitelist entries, while simultaneously blocking the malicious attempts to bypass whitelisting.

A helpful approach can be considering the vector by which new executables are introduced to the system intentionally and also by adversaries.

Acquiring Innocuous Binaries

- Do desktops download their own patches?
 - They shouldn't – for both performance and security reasons
- Do servers download executables directly?
 - Oh calculator upside-down 1134 no. They better not!
- Most endpoints should only ever receive new/updated executables from the patch management solution
 - Further, this code should, hopefully, be signed by the original vendor

Continuous Monitoring and Security Operations

67

Acquiring Innocuous Binaries

How do our users' systems obtain executables in the first place?

Do desktops download their own patches from the Internet? They shouldn't for both performance and security reasons. For example we do not want 10,000 people all trying to download the same 100 MB installer or update.

What about our servers, should they be downloading their own executables from the Internet? Not a chance.

The overwhelming majority of endpoints should only ever receive brand new or potentially updated binaries directly from the patch management solution we have in-house. If the executable is new, then ideally the installation will be deployed by our patch management or systems management solution.

Evil Executable Propagation

- After initial compromise, how do adversaries get evil executables onto boxes?
 - E-mail attachment (not anymore)
 - Download via HTTP/HTTPS
 - Download via TFTP/FTP
 - Download via DNS
 - Download via SMB
 - Download via whatever you allow outbound
 - Pivoted distribution from compromised host
 - Removable Media (USB)

Continuous Monitoring and Security Operations

68

Evil Executable Propagation

The expectation is that our systems will simply get updated software via the patch management or system management (perhaps SCCM). So, how do all those potentially evil executables make their way to our systems?

- E-mail attachment (not anymore)
- Download via HTTP/HTTPS
- Download via TFTP/FTP
- Download via DNS
- Download via SMB
- Download via whatever you allow outbound
- Pivoted distribution from compromised host
- Removable Media (USB)

Identification of Source

- Via the network, we could potentially detect/prevent the non-innocuous executable propagation
- Alternate Data Stream zone identifier that indicates the network "zone"
 - Local Computer - `Zone.Identifier:$Data == 0`
 - Local Network - `Zone.Identifier:$Data == 1`
 - Trusted - `Zone.Identifier:$Data == 2`
 - Internet - `Zone.Identifier:$Data == 3`
 - Restricted - `Zone.Identifier:$Data == 4`
- To find all files with the `Zone.Identifier` ADS
`C:\> dir /R /s | find "Zone.Identifier"`

Continuous Monitoring and Security Operations

69

Identification of Source

One interesting way to potentially identify the source of an executable on Windows NTFS partitions is through the `Zone.Identifier` Alternate Data Stream (ADS). This is an alternate data stream automatically attached to a file by Windows. The point of the `Zone.Identifier` is to indicate the zone of trust from which the particular file was acquired.

An extremely interesting aspect of ADS is that they can follow a file wherever it moves (as long as it moves from one NTFS-supporting source to another). What this means is that an executable sourced from the Internet that gets dropped into Windows/System32 could stand out like a sore thumb, if you know how to look at these.

On recent versions of Windows, you can simply use `dir /r` to see ADS

```
C:\> dir /R /s | find "Zone.Identifier"
```

Subsequently **more** or **notepad** could be used to actually view the ADS.

<http://blogs.technet.com/b/askcore/archive/2013/03/24/alternate-data-streams-in-ntfs.aspx> (http://cyber.gd/511_332) QR



Zone.Identifier

The screenshot shows a Windows Command Prompt window titled "Administrator: Command Prompt". The user is in the directory `d:\Users\Apollo\Downloads`. They run the command `dir /r packets.xls`, which shows a file named `packets.xls` with an alternate data stream (ADS) named `Zone.Identifier:$DATA`. The file size is 235,008 bytes. The user then runs the command `more < packets.xls:Zone.Identifier:$DATA`, which displays the contents of the ADS: `[ZoneTransfer]` and `ZoneId=3`. Annotations with arrows point from text boxes to the file name and the ADS path in the command.

```
Administrator: Command Prompt

d:\Users\Apollo\Downloads>dir /r packets.xls
Volume in drive D is Data2
Volume Serial Number is 721B-95BC

Directory of d:\Users\Apollo\Downloads

03/08/2014  05:58 PM                235,008 packets.xls
                                26 packets.xls:Zone.Identifier:$DATA
1 File(s)                235,008 bytes
0 Dir(s)      128,744,681,472 bytes free

d:\Users\Apollo\Downloads>more < packets.xls:Zone.Identifier:$DATA
[ZoneTransfer]
ZoneId=3

d:\Users\Apollo\Downloads>
```

Zone.Identifier

Above we see an example of using `dir` to see the Alternate Data Stream (ADS) exists. Afterwards the `more` command is used to determine the contents of the ADS.

The commands run were:

```
C:> dir /r packets.xls
```

```
C:> more < packets.xls:Zone.Identifier:$DATA
```

Whitelisting Administrative Overhead

- Operations and Maintenance of application whitelisting can be significant
- Plan for the pain
- Failure to plan for the pain often leads to failed or reduced capability deployments
- A phased deployment can be useful to reduce the initial pain
- We employ a three-phase methodology

Continuous Monitoring and Security Operations

71

Whitelisting Administrative Overhead

While application whitelisting can be a huge boon to increasing the overall security posture of our endpoints, it is not without its own inherent difficulties.

Application Whitelisting typically requires significant operations and maintenance dedication. While the initial build out will typically be the most onerous part of the task, many organizations fail to realize the extent of the ongoing burden.

To better facilitate moving toward a long-term sustainable application whitelisting environment we posit a three-phased approach.

Phase 0: Whitelist Building

- Goal: Determine the authorized executables for your organization/systems
- Potential starting points:
 - National Software Reference Library (NSRL)
 - Capture all executables on fielded systems
 - Capture all executables on pre-fielded image
- Each of these approaches has advantages and drawbacks that should be understood

Phase 0: Whitelist Building

The very first phase, Phase 0, involves the initial building of the application whitelist itself. Though conceptually simple, caution should be exercised here with the approach taken.

Over the coming slides, we will discuss the advantages and disadvantages associated with using the National Software Reference Library, capturing all executables on fielded systems, and capturing all executables on pre-fielded systems. Each of this has pros and cons when serving as the source of initial whitelisting.

NSRL RDS (Reference Dataset)

- **Advantage:** Easy to gather binaries without having to touch those icky deployed systems
- **Disadvantage:** Does this generic list include all the software your organization uses
- **Disadvantage:** Only updated four times/year
- **Disadvantage:** Will not include your custom applications

Continuous Monitoring and Security Operations

73

NSRL RDS (Reference Dataset)

NIST maintains the National Software Reference Library (NSRL). The purpose of the NSRL is to collect and maintain known files from software and operating systems and provide them as a Reference Dataset (RDS). The expected way that the RDS gets used/consumed is to ease the burden when reviewing files on an acquired system for forensics.

Regardless of the anticipated use-case, we can also leverage the NSRL RDS to provide a starting point for our whitelist. There are advantages and disadvantages to this approach.

Advantage: Easy to gather binaries without having to touch those icky deployed systems

Disadvantage: Does this generic list include all the software your organization uses

Disadvantage: Only updated 4 times/year

Disadvantage: Will not include your custom applications

<http://www.nsrl.nist.gov> (http://cyber.gd/511_320) QR



Fielded-System Executables

- **Advantage:** Unlike NSRL approach this approach can identify binaries used in your org that are legit, but not typical of every org
- **Disadvantage:** What if the system has "after-market" user binaries?
 - Hopefully this isn't possible, but it likely is
- **Disadvantage:** What if the system already has "after-market" malicious binaries?
 - That is now your known-good EVIL

Continuous Monitoring and Security Operations

74

Fielded-System Executables

One of the most significant disadvantages of leveraging the NSRL RDS is that it is necessarily not tailored at all to your organization's particular application environment, but rather is completely generic.

The other end of the spectrum for Phase 0 development of the whitelist involves capturing representative binaries from fielded-systems that are currently known working. Naturally, there are both advantages and disadvantages to this approach.

Advantage: can identify binaries used in your organization that are needed, but not necessarily found in every other organization

Disadvantage: the system could have "after-market" user binaries that have not been vetted/approved

Disadvantage: the system could already include malicious binaries, now whitelisted

Pre-Fielded System Executables

- **Advantage:** Like the fielded-system, it includes software leveraged in your organization
- **Advantage:** Unlike the fielded-system, the unfielded image is less likely to have “after-market” user binaries or malware
- **Disadvantage:** This approach is limited by how strong your builds are
 - We have already hammered home the importance of a strong baseline security configuration

Pre-Fielded System Executables

Between using the NSRL RDS on one end of the spectrum and fielded systems on the other end of the spectrum exists the possibility of using pre-fielded system executables.

This approach too has its own advantages and disadvantages when considered as the initial source of whitelist.

Advantage: Like the fielded-system, it includes software leveraged in your organization

Advantage: Unlike the fielded-system, the unfielded image is less likely to have “after-market” user binaries or malware

Disadvantage: This approach is limited by how strong your builds are

Hybrid Approach

- Start with captures of binaries that are necessary for the business
 - Prefer pre-fielded system executables, but only if that is realistic for your environment
- Have a process for automatically whitelisting patches/updates from known vendors (kinda like NSRL)
 - Ensure this capability is easy to reproduce or WL vendor provided
- Have a process for easily whitelisting binaries for individual users or, preferably, groups of users
 - Example: Former students of Ed Skoudis are allowed to use netcat
- Ensure that someone must approve (and put their butt on the line) for these exceptions

Continuous Monitoring and Security Operations

76

Hybrid Approach

The best approach for your organization might not be exclusively any one of the three previously identified approaches, but rather a combination of multiple approaches with a dash of ongoing customization and exceptions being anticipated.

One potential hybrid approach is to start with capturing executables on pre-fielded systems, or perhaps fielded system executables if the initial builds aren't that strong.

Then establish a process for managing the necessary whitelist updates that come from installation of patches and updates of the whitelisted binaries.

Also establish a process for adding custom whitelist files for specific users or, preferably, groups of users, on an as needed basis.

Phase 1: Targeted Detection

- Goal: Test the efficacy of your initial whitelist
- The targeted detection phase will help to ensure that the application whitelist isn't forever set to detect-only mode
 - While application whitelist in detect-only mode is still extremely useful, the prevention capabilities offered by whitelisting are significant
- Phase 1 simply runs the configured application whitelisting tool in detect-only mode to identify to investigate any false (or true) positives

Continuous Monitoring and Security Operations

77

Phase 1: Targeted Detection

After building the whitelist we will initially deploy it in a mode that supports targeted detection. The basic idea of phase 1 is to have a defined period of time in which the application whitelisting capability will be configured in detect-only mode.

Note that although application whitelisting in detect-only mode can still be a significant security posture improvement, there are important wins to actually employing the prevention capabilities of the application whitelisting product.

The goal of phase 1 is to help tune the application whitelist by looking for especially false positives. In this case, false positives are those binaries that are disallowed from executing even though they are necessary for the business.

True/False Positive

- While running in detect-only mode, you almost certainly will find some alerts for executables not on the list
- Scrutinize, rather than blindly whitelisting, any executable that is not already on the list of approved applications
 - False Positive: Business necessary code not already whitelisted
 - True Positive: Unnecessary, or evil, executable not on whitelist (read: possibly compromised host)
- Every organization I have worked with has uncovered preowned hosts via this project
- Almost every organization that skips the detection phase (what we call Phase 1) and jumps to blocking ends up with at least a partially failed deployment

Continuous Monitoring and Security Operations

78

True/False Positive

The easiest way to diminish the efficacy of your application whitelisting product is to allow folks to easily whitelist without additional oversight.

Each and every application whitelist alert (or block in Phase 2) must be carefully reviewed to determine whether this should have been blocked or not. Careful review is absolutely necessary.

Every organization I have ever worked with that went through this process in a diligent fashion found evidence of already compromised systems that were previously unknown to the organization.

Phase 1 Duration

- **Question:** How long should you spend in this detect-only phase of deployment?
- **Answer:** As long as it takes
- Certainly stay in phase 1 through a solid update cycle of major applications (in some cases this could be 3-6 months)
- Don't jump too early to the next phase or you risk potentially turning your whitelisting project into shelfware
 - Especially true for complex organizations with significant variance between endpoint application requirements
- Premature phase 2 could also render the application whitelisting project forever detect-only
 - Which can still be a significant security capability

Continuous Monitoring and Security Operations

79

Phase 1 Duration

How long should we stay in Phase 1, detect-only mode? It depends is a true, but not terribly helpful answer. Another not terribly helpful answer, as long as it takes.

Perhaps one of the best ways to consider the answer is to understand the disadvantage of moving too quickly into phase 2. Too rapid a migration to phase 2 often results in false positives, blocking execution of legitimate executables, which I have seen on numerous occasions render the entire whitelist forever detect-only mode in production.

The main goal of phase 1 is not simply to find things overlooked on the initial build of the whitelist. Rather, the primary goal is to establish solid processes that can allow the whitelist to be effectively managed over time.

Phase 2: Strict Enforcement

- Goal: Find/review systems that are trying to execute binaries not already on the whitelist
- Wait a second, I thought the goal was preventing execution of binaries not on the whitelist
 - That is great too, but detection is an even more important capability
- Strict enforcement now finally has us actually blocking unknown executables from running

Continuous Monitoring and Security Operations

80

Phase 2: Strict Enforcement

Some organizations never seem to make it to Phase 2, strict enforcement. Or they jump to phase 2 prematurely and end up being thrown back down to phase 1, typically for a rather protracted period.

While many see the main goal of strict enforcement to be the prevention of non-whitelisted executables from running, I see that as an ancillary benefit. The main goal, from my vantage point, is to find and review any systems that are attempting to execute binaries not already on the whitelist.

Assuming ample time and attention was devoted to phase 1, then phase 2 can provide incredibly valuable detective capabilities.

Blocking -> Detection

- Preventing execution of unknown binaries is a win
- Don't stop with prevention though or you will miss out on a significant security boon
- Determining why an unknown binary attempted execution is even more important than the actual blocking
 - Highly actionable exception/detect
- Very often the block will indicate a compromised endpoint
 - Necessarily the block is something requiring review

Continuous Monitoring and Security Operations

81

Blocking -> Detection

The ostensible goal of application whitelisting is to prevent the execution of unknown or untrusted binaries. An even more important aspect is to determine what allowed the executable to make it to the system in the first place, and, further whether this binary is needed.

Hopefully, the blocked executable is actually a trusted binary that either was not previously identified for inclusion on the list, or represents an update to an already included binary. In either case, our process has failed us, as we need to be able to get ahead of attempted execution of trusted binaries.

Naturally, the other alternative is that the binary is not in fact trusted.

Ultimately, whether ultimately a trusted or untrusted binary, we need to do root cause analysis to determine, if possible, how the binary got on the system in the first place.

Trusted Binaries

- Recall how the application whitelisting tool determines "known good"
 - Filename
 - File Location
 - Integrity Hash
 - Code Signature
- We know, and adversaries know, methods employed to determine "known good"

Trusted Binaries

Recall that one of the initial points of discussion about building the whitelist focused on the integrity of the whitelist. How do we initially identify trusted binaries, and then also what do we verify to ensure the binary presented for execution is in fact the trusted binary it purports to be.

Common ways of identifying binaries in the whitelist include:

- Filename
- File location (path)
- Integrity Hash (MD5, SHA1, SHA256)
- Code Signature (publisher, digital signature)

These are commonly used means of identifying binaries as trusted, and are also known to adversaries. So consider how they might attack these.

(Previously) Trusted Binaries

- **Filename** – Adversary uses trusted filename
- **Location** – Adversary drops file into trusted location
- **Filename + Location** – Adversary trojanizes/replaces trusted binary with evil
- **Hash** – Possible hash collision, which is not a very likely scenario (especially with SHA256)
- Hash is the strongest standard approach
 - However, even the hash-based approach is not without potential bypass

Continuous Monitoring and Security Operations

83

(Previously) Trusted Binaries

Consider some of the various ways that adversaries might attempt to circumvent our approach to whitelisting.

Filename – Adversary uses trusted filename

Location – Adversary drops file into trusted location

Filename + Location – Adversary trojanizes/replaces trusted binary with evil

Hash – Possible hash collision, which is not a very likely scenario (especially with SHA256)

Though, of these, hash is certainly the strongest approach, but it can potentially be bypassed, though not likely through hash collisions, assuming a strong hashing algorithm.

Hash Bypass

- How can adversaries bypass the executable hash integrity check?
- Rather than putting executable content on the hard disk, adversaries inject executable content into running memory
- Standard method injecting a DLL into a running process
 - Though less intrusive methods are possible too (reflective DLL injection)
- Effectively adds executable content to an existing trusted binary
- This is the most significant way to bypass application whitelisting capabilities
- Check out Jake Williams' webcast on Code Injection

Continuous Monitoring and Security Operations

84

Hash Bypass

So, how can adversaries bypass something like SHA256? The adversaries will not bypass it via the obvious hash collision approach that could possibly be attempted against a weaker hashing algorithm.

One means to gain execution capabilities is to alter executable code after it is already running. There are various methods for code injection.

Jake Williams (@malwarejake), an instructor of both advanced exploit development and advanced forensics classes with SANS, has a webcast in which he explores code injection, “50 Shades of Hidden - Diving deep into code injection”¹

[1] <https://www.sans.org/webcasts/50-shades-hidden-diving-deep-code-injection-96665>
(http://cyber.gd/511_333) QR



Advanced Application Whitelisting

- More advanced application whitelisting is required to deal with executable content added to running memory
- Not all vendors will provide this capability
 - Ask them how they handle the memory injection scenario
- Typically application whitelisting that comes as part of a larger endpoint protection suite will lack this capability

Continuous Monitoring and Security Operations

85

Advanced Application Whitelisting

Attempts to counter the potential for application whitelisting bypass are more likely to be found in standalone commercial application whitelisting products.

Though hopefully this will change, my experience with application whitelisting (also possibly called application control) functionality offered as an element of a larger endpoint security suite does not include much capability for countering techniques like code injection.

However, do not let perfect be the enemy of good. Simply because a product can be bypassed, which will necessarily be the case, does not warrant foregoing the significant protections it does afford us.

Linux AppArmor

- AppArmor adds Mandatory Access Control (MAC) capabilities to many Linux distributions
 - Included by default in Ubuntu and OpenSUSE
- Includes application whitelisting
 - Enforce mode – enforce policy, log violations
 - Complain Mode – auditing only, log violations
- It is path-based
 - Not as secure as whitelisting that uses file hash-based restrictions

Continuous Monitoring and Security Operations

86

Linux AppArmor

The following Linux distributions support AppArmor:

- Annvix
- Arch Linux
- Debian
- Gentoo
- Mandriva
- openSUSE (integrated in default install)
- Pardus Linux
- PLD
- Ubuntu (integrated in default install)¹

[1] http://wiki.apparmor.net/index.php/Main_Page (http://cyber.gd/511_59) QR



Software Restriction Policies

- AppLocker, discussed next, provides Microsoft's best approach to Application Whitelisting
- However, AppLocker is unavailable for OS prior to Windows 7 (which you should not have) and Windows Server 2008 R2 (which you might still have)
 - AppLocker might also not be available even on more modern versions of Microsoft Operating Systems if you do not have a sufficiently expensive license
- In those circumstances, Software Restriction Policies could provide some free application whitelisting capabilities
 - Though far less feature complete than AppLocker
 - Which is far less feature complete than pure-play application whitelisting products

Continuous Monitoring and Security Operations

87

Software Restriction Policies

One of the first attempts at what we now term application whitelisting comes in the form of Microsoft's Software Restriction Policies (SRP)¹. Though many think of AppLocker when considering Microsoft's approach to application whitelisting, SRP still exists and is applicable to a larger range of versions of Windows.

If you have desktop or server systems prior to Windows 7 and Server 2008 R2, respectively, then AppLocker is unavailable. Additionally, even if you have newer versions of Windows, AppLocker could still be out of reach depending upon the particular license you have.

While SRP is an option for modern systems, if your organization is serious about application whitelisting and is concerned about management overhead then AppLocker is certainly the preferred Microsoft provided approach.

If you plan to leverage AppLocker but still have downlevel clients or servers that will move from SRP to AppLocker through replacement, then you can safely have both SRP and AppLocker served via Group Policy. On systems that support AppLocker, it will take precedence over SRP even if GPO precedence would dictate otherwise².

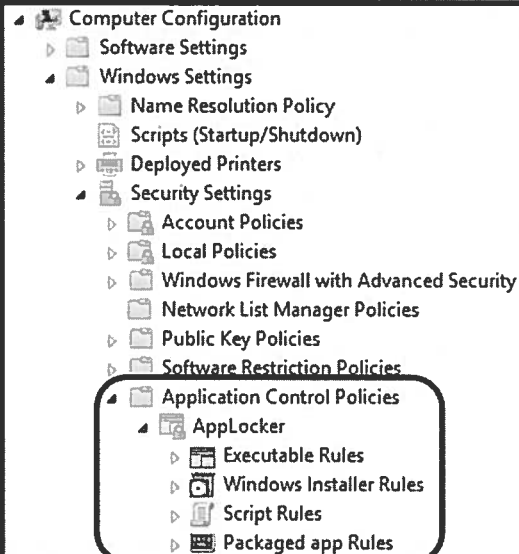
[1] <http://technet.microsoft.com/en-us/library/hh831534.aspx> (http://cyber.gd/511_334)
QR

[2] See <http://technet.microsoft.com/en-us/library/ee791851.aspx> (http://cyber.gd/511_336)



AppLocker

- Microsoft provides application whitelisting capabilities for *free*
 - Free, if you have already purchased the most expensive version of their OS
- Not as robust/full-featured as most dedicated application whitelisting products
 - But, it also might not cost anything
 - And does not require additional software or management infrastructure



Continuous Monitoring and Security Operations

88

AppLocker

Though considered the successor to Software Restriction Policies, AppLocker¹ can actually sit side-by-side with SRP. See previous page for discussions of comingling AppLocker and SRP.

There are some extremely important differences² between the functionality afforded by AppLocker and that of SRP that should be understood if the decision of which to deploy is a business decision. However, if you have licenses that allow for the user of AppLocker rather than SRP, then there is no question, you should absolutely prefer AppLocker over SRP.

AppLocker's most compelling feature for most organizations is the low low price, FREE. It is free assuming that you have already purchased the highest level of licenses for your desktop OS. Another significant advantage of AppLocker is that it is built into Windows and therefore does not require an agent to exist on the system or an additional management server/console to support and learn.

Management of AppLocker is achieved through local, or more likely, domain group policy and inherits all of the benefits and challenges already inherent in Group Policy.

AppLocker is not a full-featured replacement for a dedicated third-party application whitelisting product, but, if you already effectively own it, then AppLocker does provide significant benefits.

[1] <http://technet.microsoft.com/en-us/library/hh831409.aspx> (http://cyber.gd/511_335)

QR

[2] [http://technet.microsoft.com/en-us/library/ee424371\(v=ws.10\).aspx](http://technet.microsoft.com/en-us/library/ee424371(v=ws.10).aspx) (http://cyber.gd/511_337)



AppLocker Phase 0: Rule Creation

- Auto Generate rules from reference system
 - Should not be a currently fielded (read: compromised) system
- Deny hash rules from reference system by blocking everything referenced in a folder (hat tip to @JasonFossen and #SEC505)
 - Old versions of applications
 - Collected malware
 - Hacking tools
 - Executables discovered during an incident

AppLocker Phase 0: Rule Creation

As discussed previously, phase 0 involves building out the whitelist. AppLocker supports the automatic creation of default rules, which typically are associated with ensuring Windows still functions correctly.

Additional capabilities include the ability to leverage a reference system for building out both allow and block rules.

A technique suggested by Jason Fossen (@JasonFossen) in the highly recommended SANS SEC505: Securing Windows with the Critical Security Controls¹ (#SEC505) class, involves maintaining a reference system for AppLocker allow and deny rule creation. A specific example of employing this technique involves creating a folder that can serve as a block rule point of reference.

As new malware, hacking tools, or outdated versions of apps are identified they can be put in the folder and updated block rules can be easily generated and disseminated throughout the domain via GPOs.

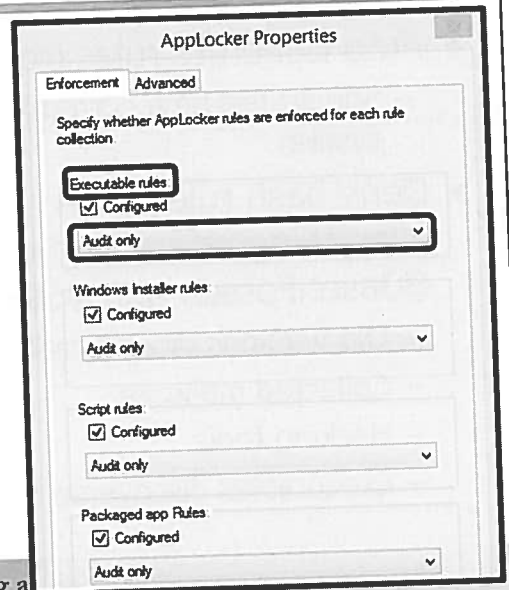
[1] <http://www.sans.org/course/securing-windows> (http://cyber.gd/511_338) QR



AppLocker Phase 1: Audit Only

- AppLocker should initially be implemented in "Audit Only" mode
- Anything that would have been blocked will result in an event log being cut
- Look for Event ID 8003 in the logs to see what would have been blocked had the enforcement been enabled

Continuous Monitoring a

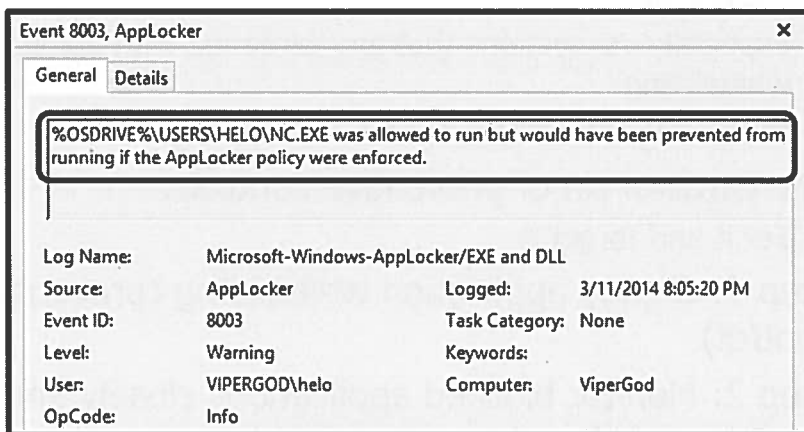


AppLocker Phase 1: Audit Only

Implementing what we described previously as Phase 1 is fairly straightforward with AppLocker. Simply choose "Audit Only" mode for AppLocker's configuration.

Recall the goal of Phase 1 is to determine what would be blocked and to understand how to establish efficient processes for handling these exceptions. AppLocker writes Event ID 8003 to the event logs whenever the AppLocker policy would have blocked execution had it not been configured in Audit Only mode.

Audit Only Mode



Continuous Monitoring and Security Operations

91

Audit Only Mode

In the slide above we see an example of Event ID 8003 being generated by AppLocker. The event in question suggests that nc.exe was allowed to run, but that it would have been blocked were AppLocker configured to enforce rather than simply audit binary execution.

"Aren't advanced attackers moving towards code and DLL injection..."

- Yes they are
 - Especially vs. systems that are hardened with application whitelisting
 - Increasing the attacker's cost == winning
- The cardinal sin of preventive controls:
 - Set it and forget it
- Step 1: Deploy application whitelisting (preventive control)
- Step 2: Monitor blocked applications closely and react in real-time (detection FTW!)

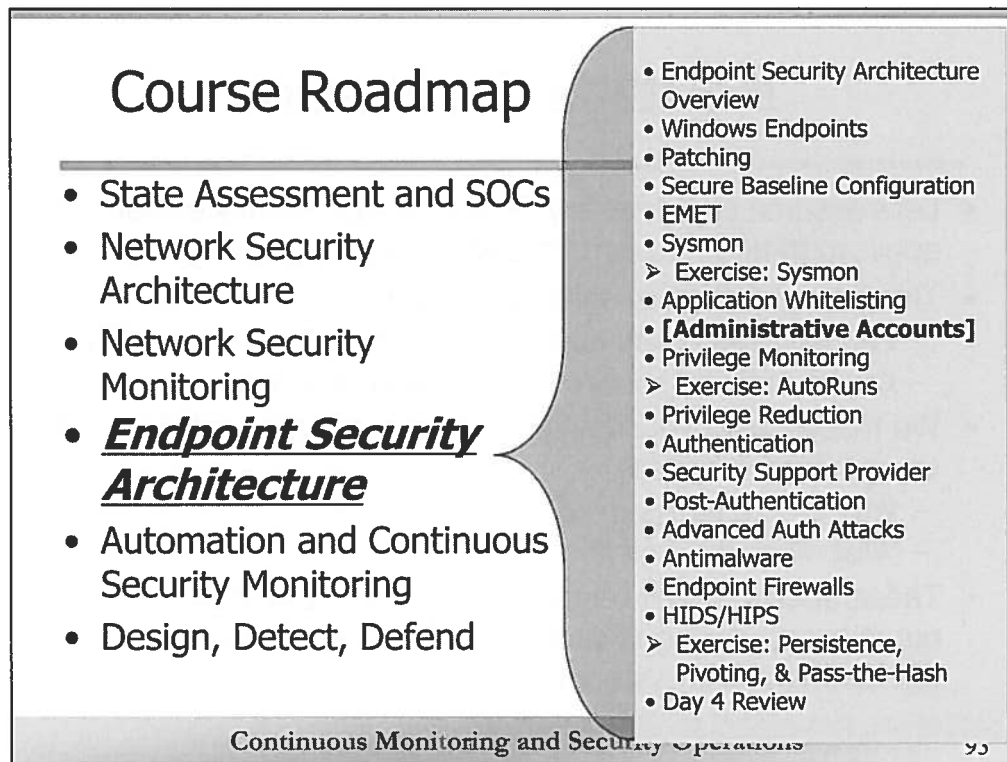
Continuous Monitoring and Security Operations

92

"Aren't advanced attackers moving towards code and DLL injection..."

Some IT people spend a lot of time and cycles shooting down great ideas, Both course authors have delivered evening talks, and mentioned that application whitelisting will (temporarily) defeat many scenarios where the attacker uses Mimikatz. Often a hand will shoot up, with the attendee eager to explain a scenario where the attacker could bypass application whitelisting.

This is true, but that normally happens after the attacker has triggered the whitelist. Your SOC, at that point, should be receiving the whitelist alert, and will trigger the incident handling process.



Course Roadmap

Our next section is Administrative Accounts.

Inevitable Exploitation

- Let's assume an adversary is able to successfully exploit an application and has remote code execution privileges
- This is exploitation in spite of focus on
 - **Patching Apps (<48 hours)** – **Patching Systems (<48 hours)**
 - **Configuration Management** – **Application Whitelisting**
- We made their task significantly more difficult, but they still compromise a system...
 - What can the adversary do?
 - What will be the impact?
- These questions will require answers to many more questions and a significant understanding of the environment and systems

Continuous Monitoring and Security Operations

94

Inevitable Exploitation

We have worked our way through four of the top five quick wins offered by the Critical Security Controls. Imagine an adversary is able to successfully exploit a vulnerable application and gains remote code execution privileges. This exploitation occurs in spite of our efforts to frustrate the adversary's ability to achieve successful exploitation through patching both applications and systems, hardened baseline security configuration, and even application whitelisting.

Is this exploitation impossible or unthinkable? No, consider it to be inevitable. Our preventive controls can and will be bypassed, which is one of the reasons we so heavily emphasize detection and response in #SEC511.

Given the exploitation, what can the adversary achieve? What will be the ultimate impact? These are certainly far from easier questions to answer, but one of the key points to consider is the privilege of the adversary.

Adversary Privilege

- One key question to help determine the capabilities of the adversary on the compromised host
- What privileges does the adversary have?
- Have they gained
 - Enterprise/Domain/Local Administrator
 - SYSTEM
 - UID 0
 - Local Service/Network Service
- Or, were we able to ensure they gained only limited (read: loser) privileges?

Continuous Monitoring and Security Operations

95

Adversary Privilege

One of the most important first considerations is the privilege gained by the adversary. If you have spent countless hours reading through Microsoft's Security Bulletins, you have no doubt seen this or very similar language, "Customers whose accounts are configured to have fewer user rights on the system could be less impacted than those who operate with administrative user rights."¹

The extent of the impact will often be tied directly to the extent of the privileges gained by the adversary. Often, not always, the privilege gained by the adversary initially has to do with the privileges of the user account on the system. Adversary privilege being tied to the user account is even more commonly the case with the increased incidence of client side exploitation.

[1] <https://technet.microsoft.com/en-us/library/security/ms14-apr.aspx> (http://cyber.gd/511_369) QR



{CSC} Privileged Accounts

- The last of the CSC First Five Quick Wins focuses on reducing account privileges
- Specifically, the Quick Win specifies a reduction in the number of users with admin privileges
- This section focuses on reducing the number of folks with highly privileged accounts
 - And also ways to limit even those highly privileged accounts we still require
- Additionally, the section will also consider aspects of authentication that can be targeted directly

Continuous Monitoring and Security Operations

96

Privileged Accounts

The last of the CSC First Five Quick Wins is concerned directly with this aspect of the risk landscape. Specifically, the goal of the quick win is to reduce the number of users with high level privileges¹. By reducing privileges generally, the number of highly privileged accounts, and also monitoring those remaining highly privileged accounts, we can be better situated to both limit the impact of exploitation, and potentially more readily identify attempts at privilege abuse.

[1] <https://www.sans.org/media/critical-security-controls/CSC-5.pdf> (http://cyber.gd/511_339) QR - page 5 specifically addresses the First Five Quick Wins



Administrative Accounts

- How many administrative accounts exist?
- Different levels of Windows admins
 - Built-in Administrator
 - Local Administrator
 - Domain Administrator
 - Enterprise Administrator
- What really constitutes an admin account?

Administrative Accounts

If the goal of this CSC Quick Win is to reduce the number of privileges, then we should be able to identify the number of admin accounts that exist.

So, obviously this includes all of those accounts that include the word “Administrator” in the account or group name.

- Built-in Administrator
- Local Administrator
- Domain Administrator
- Enterprise Administrator

Seems fairly straightforward, but what exactly constitutes an admin account beyond being overtly referred to as an admin?

How Many Administrative Accounts?

- We will baseline and monitor the administrative accounts in the CSM portion of the course
- However, we need to have a basic handle on what constitutes an “administrative account”
- Are we simply enumerating group membership?
 - That is easy, but likely not the real goal of this CSC Quick Win

Continuous Monitoring and Security Operations

98

How Many Administrative Accounts?

Once we determine what it actually means to be an administrative account then we can try to both reduce and monitor the number. Many organizations simply seem to take the obvious path of merely reducing the number of folks with Domain Administrator privileges. While this is no doubt beneficial, we need to consider what it really means to be an admin account and what all that implies.

What Does Admin Mean?

- *Admin* should mean more than just being a member of certain groups
- Imagine if I created a group called *notadmin* and then mirrored the configuration of the admin group
 - Wouldn't those accounts be just as administrative as the more colloquially named
- Nomenclature is largely meaningless, what we really care about are particular capabilities typically available to administrative accounts
 - So, to be an admin, primarily means that you have certain Windows Rights and NTFS Permissions

Continuous Monitoring and Security Operations

99

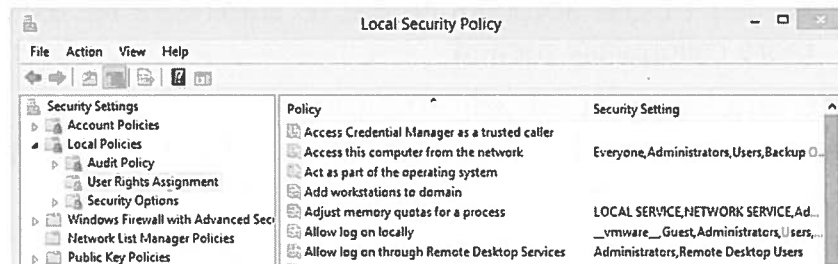
What Does Admin Mean?

To illustrate some of the difficulty of considering admin to simply mean a particular group membership, consider the following. What if I created a group called notadmin and mirrored the configuration of the group to ensure that members of this group had the same effective capabilities as that of the admin group. Would moving accounts from the traditional admin group into the notadmin group change the effective security? Not materially, though it might buy you a checkbox from certain auditors.

The name does not matter so much as the capabilities associated with the accounts. Admin implies certain default User Rights, Privileges, and NTFS Permissions. That is what constitutes being an admin, and that is what truly needs to be controlled not a naming convention.

Rights/Permissions

- Windows entitlements are fiendishly complex
- NTFS Permissions seem like simple, yet granular, file/folder permissions
 - Except these permissions exist for every registry key, printer, file, folder, and just every property of every object in Active Directory (simple, huh?)
- User Rights are more nebulous and convey user/group capabilities
 - For example: logon locally, backup files, debug programs



A later module will detail particularly critical user rights and permissions to control/monitor

Continuous Monitoring and Security Operations

100

Rights/Permissions

If you think Linux is complex, then you have clearly not spent much time working with Windows entitlements. Windows has myriad ways to convey and control entitlements. There are the seemingly simplistic NTFS permissions, that do not seem so simple when you realize that vastness of those NTFS DACLs. Every registry key, printer, file, folder, and every property of every object in AD has one.

User Rights and Privileges are even more difficult to understand and appreciate. These types of entitlements are separate and distinct from NTFS Permissions, but can certainly impact what you can and cannot do with those files, systems, and objects that are controlled via NTFS.

Built-in Administrator

- Are the built-in administrative accounts enabled?
 - If so, why?
 - Don't forget to check for the cleverly renamed admin...
- Who needs to login with administrative access that lacks an individual account with admin access?
 - Adversaries and malware
- Built-in administrator account is targeted by malware and adversaries for
 - Password guessing attacks
 - Pass-the-Hash attacks (discussed more soon)

Continuous Monitoring and Security Operations

101

Built-in Administrator

One of the simplest fixes is to consider the need of the built-in local administrator account. This account is clearly not tied to an individual user, and thereby violates some key principles of accountability. Put another way, why is the account needed? Some folks consider it to be a fail-safe in the event some sort of a disaster. If that is a consideration then consider the risk vs. reward of having the account.

This is only account that every adversary knows by name in advance of even beginning an attack. Even if you rename the account, the administrator account uses a well-known RID and so the name can easily be determined by an adversary. Additionally, this one account is not subject to account lockout by default, which means password guessing attacks are more likely to be successful and also possibly somewhat less likely to be discovered.

Perhaps the most important aspect is the default administrator account being the primary target for pass-the-hash attacks, which will be discussed further shortly.

Built-in Administrator Passwords

- Many shops leave the administrator (possibly renamed) account enabled for potential recovery
 - Bad idea, are you really going to take time to get in at this level on endpoints
- What is the password for the administrator account?
 - Probably something fairly strong – YAY!
 - Probably also synched across systems – Boo!
- Synchronized administrator accounts expose the organization to pass-the-hash style attacks

Continuous Monitoring and Security Operations

102

Built-in Administrator Passwords

As stated previously, this account is the easiest of targets for password guessing attacks. However, you very likely have a fairly strong password defined on this account because of its privilege.

Unfortunately, you are also very likely to use the same password or sets of passwords across many if not all systems internally. This is typically a feature of imaging, but, especially given the absence of salts, greatly increases the likelihood of successful pass-the-hash attacks, whereby an adversary authenticates over the network by leveraging the hash for network authentication without requiring knowledge of the cleartext password. Additional details on the pass-the-hash attacks will be provided later in the course.

Local Administrator Password Solution (LAPS)

- Microsoft released the Local Administrator Password Solution (LAPS) in May 2015
- LAPS *provides a solution to the issue of using a common local account with an identical password on every computer in a domain. LAPS resolves this issue by setting a different, random password for the common local administrator account on every computer in the domain. Domain administrators using the solution can determine which users, such as helpdesk administrators, are authorized to read passwords.*¹

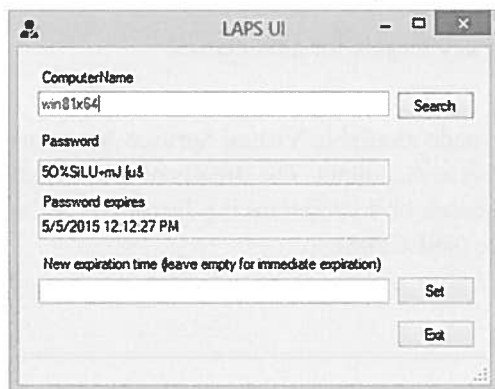
Continuous Monitoring and Security Operations

103

Local Administrator Password Solution (LAPS)

Microsoft released the Local Administrator Password Solution (LAPS), a tool for managing local administrator passwords in a domain environment.

Specified users or groups can see the password in the LAPS User Interface (see image below). The passwords are randomly generated and can be automatically changed on a schedule. The passwords are encrypted via Kerberos when sent over the network.



2



[1] <https://technet.microsoft.com/en-us/library/security/3062591.aspx> (http://cyber.gd/511_392) QR

[2] <http://www.thewindowsclub.com/local-administrator-password-solution-laps>
(http://cyber.gd/511_393)

Service Accounts

- Good old service accounts
- For many years the term "service account" was little more than a wink, wink, nudge, nudge for the auditors
- Service accounts – Those highly privileged accounts that have a user account but are not tied to a particular user and are not controlled as effectively
- Why do we need these types of account?
 - Because many applications logon without any interactive user available, but still need to run under a particular account
- Recent changes in Windows Server 2012 allow for much more robust control of service accounts than were previously available

Continuous Monitoring and Security Operations

104

Service Accounts

Another class of accounts that are very often administrative in nature are service accounts. In my experience, many organizations employed the term service account rather liberally to indicate highly privileged accounts not directly tied to a particular user and not as closely scrutinized/controlled.

The reason for having service accounts is to allow particular applications to have individualized privileges and run without require direct user interaction.

Practically, these account are often key targets for adversaries.

Recent versions of Windows have made available Virtual Service Accounts, Managed Service Accounts, and Group Managed Service Accounts. The details of these changes are beyond the scope this course. Please do your own research or consider taking Jason Fossen's (@JasonFossen) SANS SEC505: Securing Windows course (#SEC505).

LSA Secrets

- Services can leverage a standard user account rather than Local Service or Network Service
- If a regular user account (or good old domain admin), then how does the service actually authenticate?
- The password is stored in the LSA Secrets in **HKLM\Security\Policies\Secrets**
- This can be read by accounts with the Debug Programs user right/privilege
- Ouch! Please tell me these accounts only have the Log on as a service user right

Continuous Monitoring and Security Operations

105

LSA Secrets

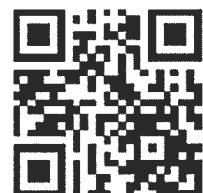
Services can, and often do, run as Local Service or Network Service. However, when someone refers to an account as a service account, they are typically implying that there is a traditional user account that is used for authentication. Given that the whole idea of having a user account log on as a service is to keep someone from having to interactively supply a password, then how exactly do the services authenticate?

The answer lies in the LSA Secrets located in **HKLM\Security\Policies\Secrets**¹.

The password for these service accounts is stored within the LSA Secrets. Any account that possesses the Debug Programs user privilege can access and decrypt this data².

[1] <http://technet.microsoft.com/en-us/library/hh994565.aspx> (http://cyber.gd/511_340) QR

[2] http://bernardodamele.blogspot.com/2011/12/dump-windows-password-hashes_20.html (http://cyber.gd/511_341)



Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- **[Privilege Monitoring]**
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 100

Course Roadmap

Our next section is Privilege Monitoring.

Privileged Account Monitoring

- Regardless of how hard we work, we will always end up with some highly privileged accounts
 - Privileged accounts are necessary, but will also necessarily be targeted
- This module will look at permissions, user rights, and privileges that are especially important to both proactively control and monitor
- Monitor closely
 - Accounts wielding these privileges unexpectedly
 - Accounts being granted these privileges

Continuous Monitoring and Security Operations

107

Privileged Account Monitoring

Perhaps we have been able to identify and even limit user accounts with high level privileges, such as admins. Wonderful, but, by necessity, we will inevitably still have some highly privileged accounts that adversaries will continually target.

The goal of this section is to understand some of those key targeted user rights and privileges, when they are likely to be used, and also monitoring for accounts using or being granted these privileges.

NTFS Permissions

- Controlling NTFS Permissions is both straightforward and cumbersome
- The main difficulty is because those NTFS Permissions requiring monitoring are less obvious than User Rights
- One approach is to emphasize data that must be tightly controlled, which might require DLP or Dynamic Access Control
- Another approach is to focus on common objects or locations targeted by adversaries

Continuous Monitoring and Security Operations

108

NTFS Permissions

Though NTFS permissions are vast and ubiquitous in Windows environments they are somewhat straightforward, and yet also cumbersome. Some of the key difficulty comes from knowing which in the panoply of NTFS permissions are incredibly important and worthy of additional scrutiny.

One approach is to focus on key data that must be protected in your organization, which might warrant DLP or Microsoft's recent Dynamic Access Control.

A second approach would be to focus on areas that are quite often targeted by adversaries.

exploit/windows/local/service_permissions

```
msf exploit(service_permissions) > info

Name: Windows Escalate Service Permissions Local Privilege Escalation
Module: exploit/windows/local/service_permissions
Platform: Windows
Privileged: No
License: Metasploit Framework License (BSD)
Rank: Great

Provided by:
scriptjunkie

Available targets:
Id Name
-- --
0 Automatic

Basic options:
Name Current Setting Required Description
-----
AGGRESSIVE false no Exploit as many services as possible (dangerous)
SESSION yes The session to run this module on.

Payload information:

Description:
This module attempts to exploit existing administrative privileges to obtain a SYSTEM session. If directly creating a service fails, this module will inspect existing services to look for insecure file or configuration permissions that may be hijacked. It will then attempt to restart the replaced service to run the payload. This will result in a new session when this succeeds. If the module is able to modify the service but does not have permission to start and stop the affected service, the attacker must wait for the system to restart before a session will be created.
```

109

exploit/windows/local/service_permissions

This slide includes a screenshot of a Windows local privilege escalation exploit that works by seeking out poorly configured NTFS permissions. In particular, this module attempts either create a service or, failing that, replace the binary associated with a service due to lax NTFS permissions. This approach can potentially allow users to gain SYSTEM level privileges.

User Rights and Privileges

- The main aspects of Admin that need to be tightly controlled and closely monitored are the significant User Rights and Privileges
 - User Rights refer to logon abilities
 - Privileges refer to particular capabilities other than logon
- Thankfully, User Rights and Privileges are readily administered via Group Policy
- An exhaustive review of all user rights is well beyond the scope of this module
 - Key critical rights/privileges warrant further detailing

Continuous Monitoring and Security Operations

110

User Rights and Privileges

Compared to NTFS permissions, user rights and privileges are simultaneously more straightforward and more convoluted, which is a bit odd to suggest, and seems altogether contradictory. User Rights and Privileges are more straightforward because after review we will understand some key dangerous rights/privileges to watch carefully. However, they are also more convoluted in that it can be harder to understand what capabilities a user gains by the privilege.

User rights are associated with logon abilities, while privileges refer to particular capabilities other than logon.

Key User Rights

- Controlling User Rights can reduce the impact of account/credential compromise
- Significant User Rights requiring scrutiny
 - Allow/Deny Log On Locally
 - Allow Log On Through Remote Desktop Services
 - Deny Access via the Network
 - Logon as a Service

Key User Rights

Windows User Rights are concerned with user's abilities to logon to systems. Though straightforward, taking proactive measures with user rights can be a significant boon to security.

- Allow/Deny Log On Locally
- Allow Log On Through Remote Desktop Services
- Deny Access via the Network
- Logon as a Service

Key Privileges

- Though privileges are a bit more opaque than user rights, controlling a few privileges can greatly increase security
- Some of the most important privileges (@JasonFossen's "Maleficent Seven")
 - Debug Programs
 - Load Drivers
 - Impersonate a Client
 - Take Ownership
 - Act as Part of the OS
 - Restore Files/Directories
 - Create a Token

Continuous Monitoring and Security Operations

112

Key Privileges

Though much less simple to understand than User Rights, Privileges are no less important to our security posture. @JasonFossen refers to these below as the "Maleficent Seven" in #SEC505.

- Debug Programs
- Impersonate a Client After Authentication
- Act as Part of the OS
- Create a Token
- Load Drivers
- Take Ownership
- Restore Files

Persistence

- As discussed on day 1, after gaining access and privileges, adversaries desire to maintain their access
- Persistence is the term used for keeping this access, and implies surviving
 - Reboot,
 - Patching the initial vulnerability,
 - Switching users, or even
 - Simple file deletion
- The primary importance of some privileges just discussed is their tie to adversaries means to persist
 - To be able to persist with high-level privileges represents an even higher goal for the adversary

Continuous Monitoring and Security Operations

113

Persistence

One of the primary post-exploitation tasks an adversary will perform is an attempt to achieve persistence. Many of the privileges just discussed are associated with adversaries attempting to gain persistent highly privileged access to a compromised system.

Exploitation, especially in the modern world of primarily client-side exploitation, can be difficult and often requires a degree of social engineering. Once an adversary has successfully compromised a victim, often through some form of convincing the victim to click a link, render a website, or open a crafted file, they will not want to have to exploit the victim again. In fact, their likelihood of successful exploitation will typically decrease, and could well require a new campaign or renewed effort.

To preclude constant need for re-exploitation, the adversary will seek to acquire persistent access to the victim.

ASEPs

- Reboot survival generally means automatically (re)starting malware
- There are **many** different ways to have a binary automatically execute on Windows
 - These means to automatically execute are referred to as ASEPs or **Auto-Start Extensibility Points**
- Some ASEPs are well-known and easily understood
 - HKLM\...\CurrentVersion\Run – Services
 - HKCU\...\CurrentVersion\Run – Scheduled Tasks
 - Start Menu\Programs\Startup – Drivers
- Other ASEPs are a bit more obscure and require digging to understand how they operate

Continuous Monitoring and Security Operations

114

ASEPs

The most common means to survive a reboot on a Windows system is to ensure that the evil will simply startup as the system does. There are a tremendous number of different ways that code can be automatically executed on a Windows system. These various means of automatic startup are referred to as ASEPs, which stands for Auto-Start Extensibility Points. Some of these are quite well-known to most technical security professionals, while others are admittedly obscure.

Monitoring for changes to ASEP entries is a great way to detect adversaries attempts to achieve persistence, which could also serve as an indicator of a more significant degree of compromise.

Autoruns

- Sysinternals tool that analyzes many different ASEPs
- GUI (autoruns.exe) or command line (autorunsc.exe) version
 - Command line version can be easily scripted
- Includes the ability to compare/diff two reports to quickly highlight changes
- Can be configured to verify code signing signatures and also produce file hashes
- Added VirusTotal integration in early 2015
 - This feature is quite handy and powerful

Autoruns

The most well-known tool for investigating ASEPs is Autoruns from Sysinternals¹. This tool, which has been around for some time and continues to be updated, exposes a large number of different ASEPs. While many security professionals have at least a passing familiarity with. The tool is most commonly used via the GUI, and can even be run directly from the Sysinternals Live site. However, an incredibly useful way to leverage Autoruns is via the command line version autorunsc.exe.

Autoruns supports saving a report as well as comparing two reports. This comparison can be hugely beneficial from a baselining standpoint, and if historical records are maintained can be an extremely useful means of determining not only what new autostart entries exist, but also when they first appeared. Beyond the GUI comparison capabilities, the command line version offers a delimited text-based output that can be parsed easily from the command line.

Some further aspects of Autoruns can allow for more detailed investigation of entries. The “verify code signatures” option in Autoruns means that the tool will look at the code that has a listed publisher and ensure that the code has a valid signature. Verified or Not Verified will be listed next to each publisher in the GUI to indicate whether the signature is valid or not.

[1] <http://technet.microsoft.com/en-us/sysinternals/bb963902.aspx>
(http://cyber.gd/511_379) QR



Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - **[Exercise: AutoRuns]**
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 110

Course Roadmap

Our next section is the AutoRuns exercise.



SEC511 Workbook AutoRuns

Exercise 2: AutoRuns

Continuous Monitoring and Security Operations

117

SEC511 Workbook AutoRuns

Please go to the 511 Exercise Workbook, section 511.4-2.

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- **[Privilege Reduction]**
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 118

Course Roadmap

Our next section is Privilege Reduction.

Reducing Privileges

- Knowing the dangers of having highly privileged accounts, and better understanding those privileges
 - We have made reasonable steps toward reducing privileges
- Unfortunately, you will inevitably make some mistakes and limit required privileges
- This section focuses on assisting the process
- If you are serious about reducing privileges, then plan to spend some time at Aaron Margosis' blog "Non-Admin, App-Compat and Sysinternals WebLog"

Reducing Privileges

Now that we understand some of the risks associated with particular privileges, let's explore tactics to help reduce them in a safe and effective manner. No doubt, you will almost certainly take away what you believe to be unnecessary privileges only to discover that they were indeed necessary.

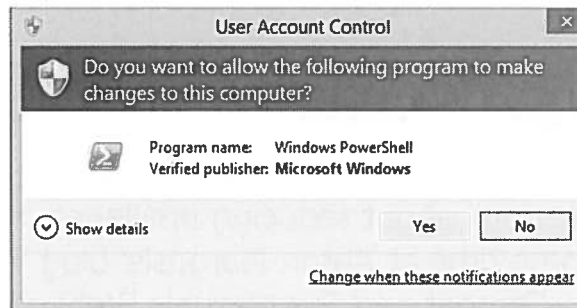
Plan on spending some time getting familiar with Aaron Margosis' blog "Non-Admin, App-Compat, and Sysinternals Weblog."¹

[1] http://blogs.msdn.com/b/aaron_margosis/ (http://cyber.gd/511_346) QR



UAC: Less (privilege) Is More (security)

- User Account Control, that much maligned security feature that came to life with Vista



Continuous Monitoring and Security Operations

120

UAC: Less (privilege) Is More (security)

One of the recent advances in controlling privileges is the much reviled User Account Control (UAC)¹ feature. UAC was released as part of Windows Vista and was notably made fun of hilariously in one of the classic “I’m a mac, and I’m a PC” ads².

[1] <http://technet.microsoft.com/en-us/library/jj574202.aspx> (http://cyber.gd/511_343) QR

[2] <http://www.youtube.com/watch?v=VuqZ8AqmLPY> (http://cyber.gd/511_342)



But I'm an Admin...

- Even when logged in with an Admin account, by default, processes will run with reduced privileges
- This behavior can cause frustration and be annoying to some, but **this annoyance is a feature, not a bug**
- Unless you expressly intend to elevate privileges, then your admin account will still run with loser privileges
 - This is a very very good thing!
- You have cowboys that like to browse the web with their Enterprise Admin account
 - When they run into that drive-by-download injected watering hole, then the adversary gains lower privileges

Continuous Monitoring and Security Operations

121

But I'm an Admin...

With UAC enabled, even though an account might actually be an administrator, their processes will, by default run with reduced privileges. While the Mac ad is funny, and folks do still find UAC (and other modern operating systems' privilege approval process) annoying, this annoyance is a strong security feature. Further, the annoyance can be customized to be more or less annoying depending on the degree to which you loathe your users ;)

The most important aspect of UAC is that, by default, unless you intentionally run most applications by explicitly elevating privileges, then they will run with lesser privileges. If these are insufficient then they can be elevated. However, most often, the process will work just fine with the lesser privileges.

The big win comes from our continuing to have more privileges than is safe. Sometimes that is necessary, but if an un-elevated application gets compromised while running the adversary does not immediately gain the full administrative privileges we might natively possess.

Loser PowerShell

```
PS C:\Windows\system32> whoami
sec511-win8\hotdog
PS C:\Windows\system32> whoami /priv

PRIVILEGES INFORMATION
-----
Privilege Name      Description                                     State
-----
SeShutdownPrivilege Shut down the system                           Disabled
SeChangeNotifyPrivilege Bypass traverse checking                       Enabled
SeUndockPrivilege    Remove computer from docking station          Disabled
SeIncreaseWorkingSetPrivilege Increase a process working set                 Disabled
SeTimeZonePrivilege  Change the time zone                         Disabled
PS C:\Windows\system32> _
```

Hotdog, a standard user account

Loser PowerShell

Above we can see a normal user account's privileges.

The commands `PS C:> whoami` and `PS C:> whoami /priv` were executed.

Admin (un)elevated PowerShell

```
PS C:\Windows\system32> whoami
sec511-win8\apollo
PS C:\Windows\system32> net localgroup administrators
Alias name     administrators
Comment       Administrators have complete and unrestricted access to the compu
ter/domain
Members
-----
Administrator
Apollo
eric
student
The command completed successfully.
PS C:\Windows\system32> whoami /priv

PRIVILEGES INFORMATION
-----
Privilege Name        Description                                     State
-----
SeShutdownPrivilege   Shut down the system                           Disabled
SeChangeNotifyPrivilege Bypass traverse checking                       Enabled
SeUndockPrivilege     Remove computer from docking station          Disabled
SeIncreaseWorkingSetPrivilege Increase a process working set                 Disabled
SeTimeZonePrivilege   Change the time zone                          Disabled
PS C:\Windows\system32>
```

Apollo an Admin defaults to the same privileges as loser, **Hotdog**...

Continuous Monitoring and Security Operations

123

Admin (un)elevated PowerShell

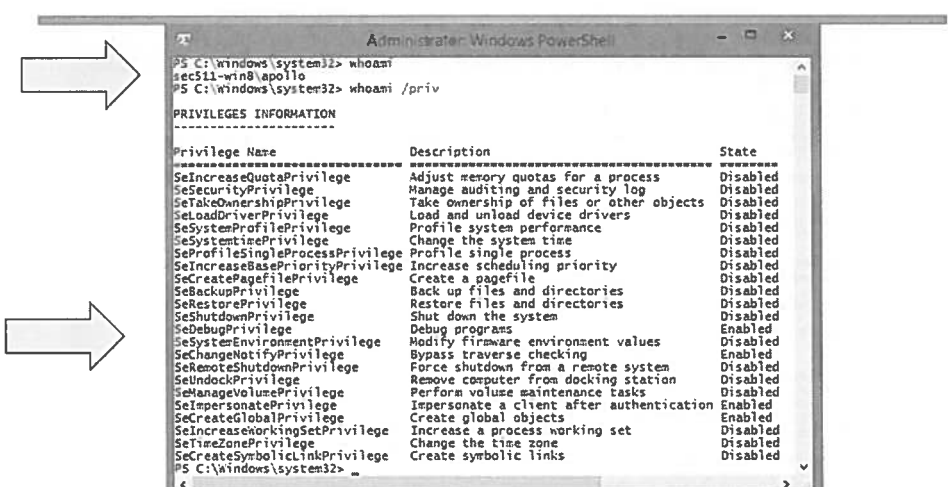
Here we can see an admin running with UAC enabled, and without having explicitly elevated his privileges.

These commands were executed:

```
PS C:> whoami
PS C:> net localgroup administrators
PS C:> whoami /priv
```

Note that the set of privileges are the same as found in that of the standard user account.

Admin Elevated PowerShell



```

PS C:\windows\system32> whoami
sec511-win8\apollo
PS C:\windows\system32> whoami /priv

PRIVILEGES INFORMATION
-----
Privilege Name      Description                                     State
-----
SeIncreaseQuotaPrivilege  Adjust memory quotas for a process           Disabled
SeSecurityPrivilege      Manage auditing and security log             Disabled
SeTakeOwnershipPrivilege Take ownership of files or other objects      Disabled
SeLoadDriverPrivilege    Load and unload device drivers               Disabled
SeSystemProfilePrivilege Profile system performance                    Disabled
SeSystemTimePrivilege    Change the system time                       Disabled
SeProfileSingleProcessPrivilege Profile single process                       Disabled
SeIncreaseBasePriorityPrivilege Increase scheduling priority                 Disabled
SeCreatePagefilePrivilege Create a pagefile                             Disabled
SeBackupPrivilege        Back up files and directories                 Disabled
SeRestorePrivilege       Restore files and directories                 Disabled
SeShutdownPrivilege      Shut down the system                         Disabled
SeDebugPrivilege         Debug programs                               Enabled
SeSystemEnvironmentPrivilege Modify firmware environment values           Disabled
SeChangeNotifyPrivilege  Bypass traverse checking                     Enabled
SeRemoteShutdownPrivilege Force shutdown from a remote system          Disabled
SeUndockPrivilege        Remove computer from docking station          Disabled
SeManageVolumePrivilege  Perform volume maintenance tasks             Disabled
SeImpersonatePrivilege   Impersonate a client after authentication    Enabled
SeCreateGlobalPrivilege  Create global objects                        Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set                 Disabled
SeTimeZonePrivilege      Change the time zone                         Disabled
SeCreateSymbolicLinkPrivilege Create symbolic links                         Disabled
PS C:\windows\system32>
  
```

Apollo running PowerShell with elevated privileges looks different

Continuous Monitoring and Security Operations 124

Admin Elevated PowerShell

Now, running with elevated privileges, we again execute `PS C:> whoami /priv` to illustrate the significant difference UAC makes.

If both of Apollo's PowerShell instances were compromised which do you think would likely have the more significant impact?

Magic Local Admin

- UAC is disabled by default for the built-in local administrator account (RID 500)
- Great, the one account everyone knows that is often synchronized across systems doesn't benefit from UAC...
- I know what you are thinking, Administrator is disabled by default on recent Windows
 - Yup, until it gets re-enabled
- Ensure UAC applies even to the administrator

Continuous Monitoring and Security Operations

125

Magic Local Admin

By default UAC is disabled for local administrator (RID 500), the one account that everyone knows by name and RID and by default is not able to be locked out.

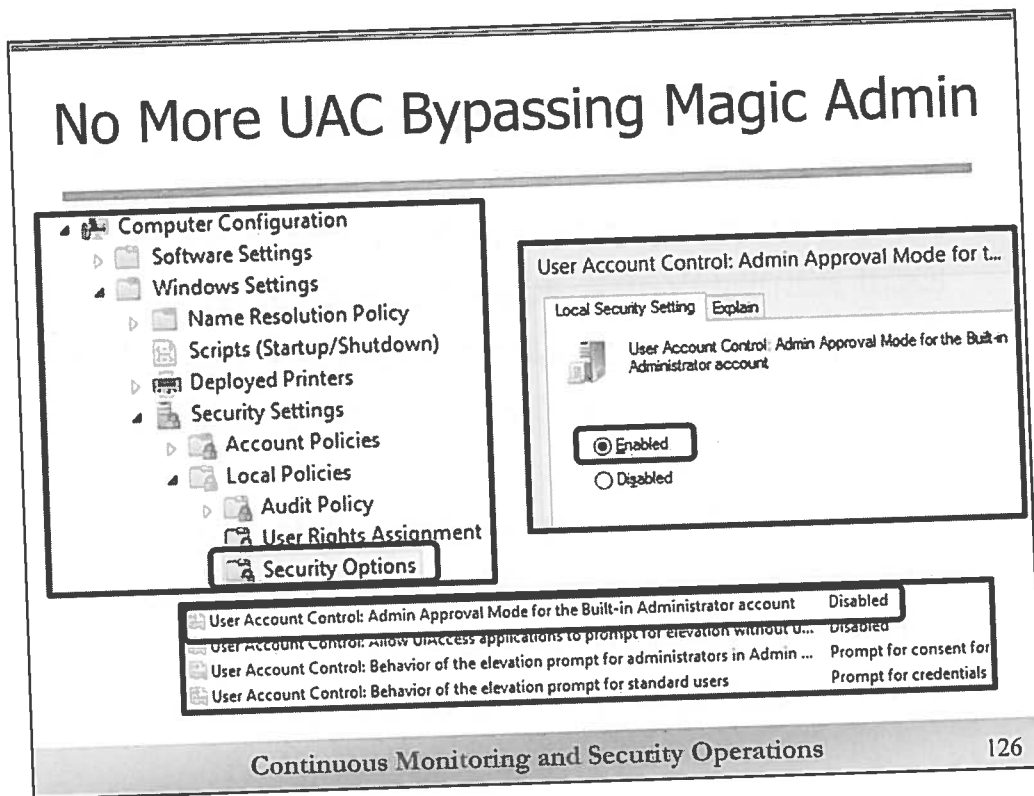
In addition to disabling the account, you should also ensure that UAC is configured to apply to the administrator account should it be re-enabled for malicious or benign purposes¹. This setting is available at: \Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options -> User Account Control: Admin Approval Mode for the Built-in Administrator account.

The default is disabled: "The built-in Administrator account logs on in Windows XP Mode, and it runs all applications by default with full administrative privileges." Changing it to enabled means "The built-in Administrator account logs on in Admin Approval Mode so that any operation that requires elevation of privilege displays a prompt that provides the administrator the option to permit or deny the elevation of privilege."²

[1] <http://technet.microsoft.com/en-us/library/jj852255.aspx> (http://cyber.gd/511_344) QR

[2] *ibid.*





No More UAC Bypassing Magic Admin

The screenshots above show how to ensure that the built-in administrative account also gains the security features afforded by UAC. The setting is found in Group Policy under Computer Configuration->Windows Settings->Security Options. Look for "User Account Control: Admin Approval Mode for the Built-in Administrator account."

By default this setting is disabled. Enabling this option will ensure that the built-in administrator account has the same experience as other administrators on the system.

Process Monitor

- Microsoft Sysinternals' Process Monitor proves extremely useful when attempting to reduce privileges
- Very often poorly coded apps that "require admin" or "require UAC disabled" simply fail because of a particular registry, file, or folder security issue
- Process Monitor proves helpful determining the cause of these access denied conditions

Continuous Monitoring and Security Operations

127

Process Monitor

The path of least resistance is the approach most often taken by tech support, especially third-party vendors. This often results in many products suggesting that users have to be administrators or UAC must be disabled.

While each of these could be the case, often vendors are simply being lazy (read: providing efficient service). On numerous "admin required" occasions I have found that simple NTFS permission changes on files, folders, or registry keys has resulted in admin not being required. However, the onus was on me to discover this fact.

Process Monitor from Microsoft Sysinternals is hugely helpful in identifying these simple issues that require permission changes.

Process Monitor: Access Denied

- Below we see Helo trying unsuccessfully to open a suspicious file

The screenshot shows the Process Monitor application window with the following data:

Time of	Process	PID	Operation	Path	Result	Detail	User
8:33:05.4	cmd.exe	3596	QueryDirectory	C:\Windows\System32\batl\hidden\Cy*	SUCCESS	Filter: Cy*, 1 CylonDetector.txt	VIPERGOD\helo
8:33:05.4	cmd.exe	3596	QueryDirectory	C:\Windows\System32\batl\hidden	NO MORE FILES		VIPERGOD\helo
8:33:05.4	cmd.exe	3596	CloseFile	C:\Windows\System32\batl\hidden	SUCCESS		VIPERGOD\helo
8:33:06.4	cmd.exe	3596	CreateFile	C:\Windows\System32\batl\hidden\CylonDetector.txt	SUCCESS	Desired Access: Read Data/List Directory, Synchron...	VIPERGOD\helo
8:33:06.4	cmd.exe	3596	QueryDirectory	C:\Windows\System32\batl\hidden\CylonDetector.txt	SUCCESS	Filter: CylonDetector.txt, 1 CylonDetector.txt	VIPERGOD\helo
8:33:06.4	cmd.exe	3596	CreateFile	C:\Windows\System32\batl\hidden\CylonDetector.txt	SUCCESS	Desired Access: Read Attributes, Disposition: Open	VIPERGOD\helo
8:33:06.4	cmd.exe	3596	QueryBasicInfo	C:\Windows\System32\batl\hidden\CylonDetector.txt	SUCCESS	CreationTime: 2/24/2014 8:30:30 AM, LastAccess...	VIPERGOD\helo
8:33:06.4	cmd.exe	3596	CloseFile	C:\Windows\System32\batl\hidden\CylonDetector.txt	SUCCESS		VIPERGOD\helo
8:33:06.4	cmd.exe	3596	CreateFile	C:\Windows\System32\batl\hidden\CylonDetector.txt	ACCESS DENIED	Desired Access: Generic Read, Disposition: Open	VIPERGOD\helo
8:33:06.4	cmd.exe	3596	QueryDirectory	C:\Windows\System32\batl\hidden	NO MORE FILES		VIPERGOD\helo
8:33:06.4	cmd.exe	3596	CloseFile	C:\Windows\System32\batl\hidden	SUCCESS		VIPERGOD\helo

Showing 563 of 334,920 events (0.16%) Backed by virtual memory

Continuous Monitoring and Security Operations 128

Process Monitor: Access Denied

Typically the first task after being told that an app requires admin is to summarily ignore the suggestion. If the app doesn't simply work with standard user privileges, then attempt to again run the app with user privileges having first kicked off Process Monitor.

Now we look for any messages where the result column indicates "ACCESS DENIED".

Application Compatibility Toolkit (ACT)

- Another Microsoft tool for assisting with failed attempts to reduce privileges is the Application Compatibility Toolkit
- The main purpose of the toolkit is to assist with problems when migrating applications to newer versions of Windows
- However, it can also be used to “fix” those apps that simply verify admin before attempting to run at all
- We can use the ACT to have Windows establish an environment that allows the application to run
 - Sometimes all we have to do is lie to the application in order to get it to run
- In ACT the fixes we define are referred to as shims

Application Compatibility Toolkit (ACT)

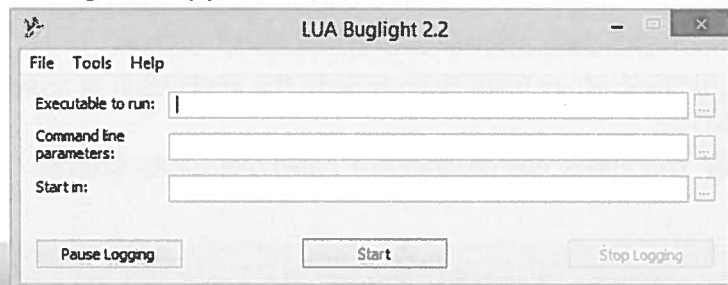
Another Microsoft tool that can prove extremely helpful when trying to limit privileges is their Application Compatibility Toolkit¹ (ACT). The primary goal of the ACT is to aid with migrating to more recent operating systems. The ACT can be used to assist with getting legacy apps to run on modern Windows OS. However, the ACT can also be leveraged for our specific purposes, and even provides the Standard User Analyzer Tool/Wizard to specifically assist with our problem directly.

[1] <http://technet.microsoft.com/en-us/library/hh825181.aspx> (http://cyber.gd/511_345) QR



LUA Buglight

- Tool by Aaron Margosis specifically targeting privilege reduction
 - Supports XP - Win8
- LUA Buglight helps identify “LUA Bugs”
 - Issues that prevent standard users from successfully running the application

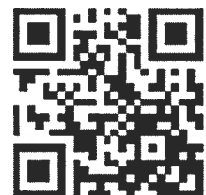


130

LUA Buglight

Though the Application Compatibility Toolkit can be used for our purposes, namely identifying issues that are preventing privilege reduction, LUA Buglight was specifically designed for this purpose. Aaron Margosis created and continues to update LUA Buglight to help identify application issues that are forcing elevated privileges for successful execution.

[1] http://blogs.msdn.com/b/aaron_margosis/archive/2012/11/28/lua-buglight-2-2-with-support-for-windows-8.aspx (http://cyber.gd/511_347) **QR**



Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- **[Authentication]**
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations

131

Course Roadmap

Our next section is Authentication.

Authentication

- Adversaries place significant value in compromising authentication credentials
 - Rightly so, if you have, or can (re)use, authentication credentials then exploitation and post-exploitation is significantly easier
- Public facing credentials are important, but the most significant are typically Windows credentials
 - Windows credentials increase the likelihood and impact of pivoted post-exploitation
- We reduced the number of folks with admin privileges
- We reduce the capabilities of admin and standard accounts
 - Still, higher and lower privileged accounts will still exist
 - there will always be both low and high privileged accounts employed in any environment

Continuous Monitoring and Security Operations

132

Authentication

An adversary will compromise a system, of this we have little doubt. One of the highest value post-exploitation targets is that of authentication credentials. This could be cleartext username/password, password hashes, access tokens, or Kerberos tickets. The abuse of legitimate credentials serves as one of the most common, and difficult to both prevent and detect, means of pivoting closer to the adversary's actual target.

Windows credentials are some of the most valuable due to the nature of Windows single-sign-on and their use internal to the organization. To ensure we have any hope of protecting these credentials and detecting potential abuse, we must develop a strong understanding of the ins and outs of authentication in the Windows world.

Passwords

- The most basic and common means to authenticate simply is to provide a password along with a userid
- Passwords have a number of issues associated with them that we have understood well for years
- Though the use of smart cards and two factor authentication has increased
 - Passwords will still be plaguing us for many years to come
- The primary focus of this section will be on Windows authentication
 - First some general password considerations will be explored

Passwords

Though we have seen increasing use of two factor authentication, especially with well-known public facing web applications, the fact remains that passwords are still the de facto. We will quickly move through some of the more basic aspects of passwords and then move on to some less straightforward aspects of these credentials.

Password Reuse/Synchronization

- Most people are fairly lazy when it comes to passwords
- Once they finally come up with a password they can remember that meets the length/complexity requirements
 - They sometimes only want to remember that one password
 - So they use it everywhere they can
- This common practice can make one breach of an unrelated app/system/organization have further reaching impacts

Continuous Monitoring and Security Operations

134

Password Reuse/Synchronization

Challenges with password reuse exist on multiple levels. First, the same or a very similar (e.g. **Password1**, **Password2**, **Password3**, etc.) password might be reused on the same system. Some systems, notably not Windows, can force users to set passwords that are sufficiently different from the previous password (and what constitutes sufficient can be defined).

Another aspect of password reuse involves users self-synchronizing the same, or very similar, password that can then be leveraged across multiple systems/applications/domains.

The compromise of one credential could have a more significant than anticipated impact if reuse occurs.

Windows Password Hashes

- Windows passwords/hashes serve as some of the most common and valuable credentials
- Widely used within enterprises for SSO to many deployed applications
 - Windows passwords/hashes falling are the first domino
- Architecturally, Windows strongly supports SSO
 - Convenient for the end users
 - Greatly increases the impact of compromise

Windows Password Hashes

Given the ubiquity of Windows in enterprises, it is not terribly surprising to find out that Windows passwords and their hashed representations represent high value targets for adversaries. Beyond the direct value to the Windows environment itself, Active Directory often serves as the primary enterprise identity provider and is very often used for aspects of enterprise single sign on (SSO).

At a fundamental level, Windows fully supports SSO, at the very least to Microsoft-provided offerings. While SSO can be a security adjuvant, it can also aid our downfall as well.

Windows a Low Sodium Architecture

- Windows static hashing algorithms do not employ salts
 - If we have the same password, then we have the same resultant hash
- The lack of salts increases the efficacy of attacks such as Pass the Hash
- Also renders Windows hashes particularly vulnerable to pre-computation brute force cracking
 - Rainbow Tables are the common term used for the pre-computation brute force attack

Windows a Low Sodium Architecture

Windows password hashes do not employ cryptographic salts. A cryptographic salt is simply a degree of randomness that is incorporated into the password hashing algorithm. As we know the principle of hashing algorithms is that the same input, in this case, the password, yields the same output, resultant hash every time the process is carried out.

same input :: same output is necessary, however, without salts, the concept of same input is much more vast than most people consider. For Windows passwords/hashes, effectively anyone in the world that has ever leveraged your exact password (same input) results in the exact same hash (same output).

Conceptually this means that if a user's password is ever cracked and the input::output stored, then that password will be cracked for any future encounter of that hash. Taken further, what if an adversary, in advance, computed every possible password and its resultant hash. Effectively the adversary would have already cracked any possible password in advance of needing that particular hash to be cracked. This is referred to as the pre-computation brute force attack, or more commonly rainbow tables.

No Salt Illustrated

The screenshot shows a Windows command prompt window with the following output:

```
mimikatz # sekurlsa::msv

Authentication Id : 0 ; 12009096 (00000000:00b73e88)
Session          : Interactive from 0
User Name        : Administrator
Domain          : VIPERGOD
SID              : S-1-5-21-2013480549-2569447298-4169852308-500

msv :
[00010000] CredentialKeys
* NTLM      : 682255653b8453cb0d8764616f81b79e
* SHA1      : 422a9e40d9a15e3b7f93dff551e4da851fd05f47
[00000003] Primary
* Username  : Administrator
* Domain    : VIPERGOD
```

Below the command prompt is a Notepad window titled "XPpass - Notepad" showing the output of the Windows Credentials Editor. It lists the Administrator account for the CYLONH8R domain with the same NTLM hash: 682255653b8453cb0d8764616f81b79e.

Two arrows point from the text "Do you see what I see?" to the NTLM hashes in both windows, highlighting that they are identical despite being from different domains (VIPERGOD and CYLONH8R).

Continuous Monitoring and Security Operations 137

No Salt Illustrated

The above slide illustrates the problem associated with a lack of salts. Here we see the local administrator account on two different systems, VIPERGOD and CYLONH8R, clearly employs the same password, given the same hash.

LM==LaMe

- The LM (LAN Manager) Hash is Microsoft's legacy password hash
- Seems purpose-built to allow fast cracking
- Key LM algorithmic FAILs
 - Only supports uppercase
 - Requires two separate 7 character strings
 - No salts
 - Not cryptographically expensive (DES)
 - Blank/Empty hash well-known
- Even though the NT hashing algorithm has long been available, LM still seems to find support for "backwards compatibility" purposes

Continuous Monitoring and Security Operations

138

LM==LaMe

There are more poor password hashing algorithms in use than strong ones, but LM, or LAN Manager, is easily one of the poorest.

Some of the key weaknesses inherent in LM:

- Only supports uppercase
- Leverages two separate 7 character strings
- No salts
- Not cryptographically expensive (DES)
- Blank/Empty hash well-known (**AAD3B435B51404EE**)

Is LM Finally Behind Us?

- Since Windows Vista Microsoft, by default, no longer stores the LM hash in the SAM file
 - Prior to Vista this was a non-default configurable option
- Unfortunately there is evidence to suggest that the LM hashes are still created and available in running memory
- The LM hash is the (attacker's) gift that keeps on giving and giving and giving
- Best way to ensure no LM hash is to employ a 15 character (or more) password

Continuous Monitoring and Security Operations

139

Is LM Finally Behind Us?

By default now (since Vista) Windows no longer stores LM hashes in the SAM. Further, we have been able to configure this option prior to Vista in local or domain Group policy. So, can we safely say that LM can go the way of the dodo in favor of this “modern” hashing algorithm NT that was released with Windows NT in the early 90s.

Sadly, LM keeps coming back. Hernan Ochoa, author of the Windows Credential Editor and also the prior Pass the Hash toolkit, discovered that the LM hash persists in RAM, even if it is not available in the SAM¹.

Passwords that are 15 characters or longer break the LM hash algorithm: the LM hash cannot be calculated. This means the simplest way to avoid usable LM hashes in the SAM or in memory is to use a 15-character or longer password.

[1] http://www.ampliasecurity.com/research/wce12_uba_ampliasecurity_eng.pdf
(http://cyber.gd/511_348) QR



NT

- The NT hashing algorithm is decidedly stronger than the LM hash approach
- Key NT wins over LM
 - Full password (up to 127) gets hashed
 - Case sensitivity persists
 - Wider character set support
- NT still FAILs on these accounts
 - No salts
 - Blank/Empty hash well-known
 - Not cryptographically expensive (MD4 based)

Continuous Monitoring and Security Operations

140

NT

Though a vast improvement over LM, NT certainly does not constitute a preferred password hashing algorithm. At the highest level the algorithm simply starts with Plaintext -> Unicode -> MD4. We do get longer possible passwords, case sensitivity, and a wider set of supported characters.

Jesper Johansson of Microsoft published a nice presentation entitled Windows Passwords: Everything You Need to Know that is available for download. While the title might overstate things a bit, the presentation does provide a fairly approachable and short guide to the world of Microsoft authentication.

<http://download.microsoft.com/download/f/4/a/f4a67fc8-c499-461d-a025-8155fb4f7a0f/Windows%20Passwords%20Master%201.5%20Handout%20-%20Jesper%20Johansson.ppt> (http://cyber.gd/511_349) **QR**



Password Storage

- Windows password hashes can end up in a number of different locations
- The local SAM serves as the expected storage location for Windows password hashes and is located in `C:\Windows\System32\config\SAM`
- The SAM only includes hashes for local accounts
- Domain hashes reside within Active Directory
- Physically the domain account hashes are located in `C:\NTDS\Ntds.dit` on Domain Controllers

Continuous Monitoring and Security Operations

141

Password Storage

There are two standard locations for Windows password hashes to live by design, `C:\Windows\System32\config\SAM` and `C:\NTDS\Ntds.dit`, with the former being on the local systems while the latter is the location of Active Directory which would contain hashes for all domain accounts.

There are many other potential nooks and crannies where passwords and backup copies of the SAM might live, but these are the expected locations.

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- **[Security Support Provider]**
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations

142

Course Roadmap

Our next section is Security Support Providers.

Security Support Provider (SSP)

- Microsoft enables several Security Support Providers (SSP)
 - The SSPs are packages that allow for different types of authentication to occur
- Transparency and single-sign-on are key Microsoft design goals for authentication purposes
- Ideally, for Microsoft, you would authenticate once and then be able to seamlessly leverage that credential throughout an environment including:
 - Active Directory (Kerberos, NTLMv1/2, LM Challenge Response)
 - Web applications (NTLM integrated authentication, HTTP Digest)
 - Remote Desktop Services
- Network-based applications leverage SSPs through a Security Support Provider Interface (SSPI)

Continuous Monitoring and Security Operations

143

Security Support Provider (SSP)

While we often think first of things like LM and NT Hashes when considering Windows authentication, under the hood Microsoft enables various Security Support Providers (SSP). The LM and NT hashes exist as a part of a larger authentication infrastructure provided by Windows.

Microsoft includes SSPs that will feel quite familiar and are naturally associated with LM/NT and what we expect from Windows authentication. However, there are also others that will leave you scratching your head with a bit of a confused look on your face.

Recall that we said previously that Windows architecturally and fundamentally supports Single Sign On, well this is where we begin to appreciate the scope of their SSO support.

SSP Impact of Single Sign On

- To facilitate ease of use and single-sign-on, Microsoft will pre-generate different forms of credentials and store it in memory
- LM and NT hashes naturally would be in running memory to support future authentication needs
- Other SSPs credentials can also be pre-calculated and stored in memory

Continuous Monitoring and Security Operations

144

SSP Impact of Single Sign On

In order to be able to facilitate efficient and transparent SSO, Microsoft must hold a number of items in running memory. As can be expected the LM (seriously) and NT hash could be needed to allow the system to authenticate across the network using LM Challenge Response (NOOOOO!!!!!!), NTLMv1 (NO!!!), NTLMv2 (If possible, no), Kerberos (more like it).

However, what about performing authentication that does not directly involve LM or NT? To the extent possible, Microsoft wants you Windows, AD, and, especially, now your Microsoft Account (formerly known as Live Account) to provide you access to all different types of resources. To keep you within their ecosystem, they want to enable the use of their credential ubiquitously.

SSP: WDigest

- Many folks first real taste of an unexpected SSP is with WDigest, which was introduced in Windows XP
- The WDigest SSP (implemented via `wdigest.dll`) exists to facilitate HTTP Digest Authentication
 - HTTP Digest is a Challenge Response authentication protocol meant to address a major deficiency in HTTP Basic Authentication
 - The primary issue with HTTP Basic authentication is sending passwords across the wire Base64 Encoded
- In order to provide this functionality, on the fly, without requiring reauthentication, Windows stores the cleartext password in a readily reversible fashion
 - Wait, they do WHAT????

SSP: WDigest

If you weren't buying the hard sell on Microsoft wants to be SSO provider for the universe then consider the following. Microsoft provides an SSP dedicated to support HTTP Digest Authentication¹. HTTP Digest is a built-in HTTP authentication scheme that was built to provide an alternative to the original HTTP authentication, HTTP Basic². One of the most notable issues taken with Basic authentication is the fact that it included simple Base64 encoded username and password sent across the wire. Digest authentication moved to a more sensible challenge-response authentication scheme that did not send credentials over the wire.

Microsoft allows for the use of Windows credentials to be leveraged for HTTP authentication. While IIS servers can support non-HTTP specific authentication methods, such as leveraging Kerberos, Microsoft wants your credentials to be widely supported. In order to pass through the HTTP Basic authentication without requiring you to supply your Windows username:password in that ugly browser popup box, Microsoft keeps necessary information accessible in running memory.

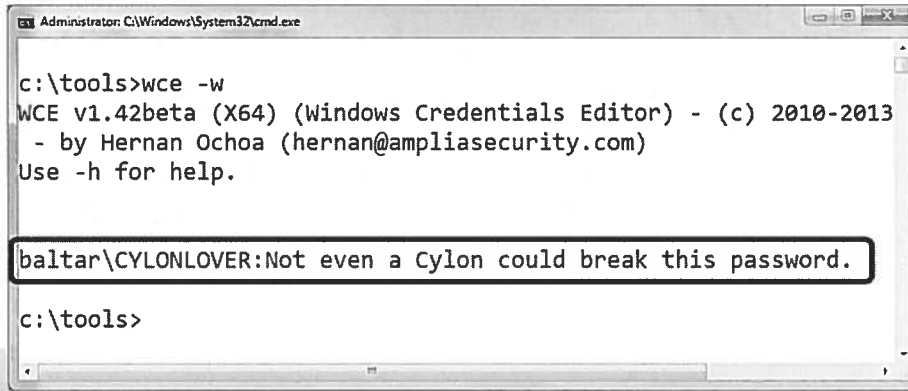
Unfortunately, your cleartext password is needed to support this pass-through authentication. Yup, you heard me right, cleartext password is in RAM to support the WDigest SSP.

[1] <http://tools.ietf.org/html/rfc2617> (http://cyber.gd/511_350)

[2] <http://tools.ietf.org/html/rfc1945#section-11> (http://cyber.gd/511_351)

WDigest FAIL

- Whether HTTP Digest is needed or not, Windows could still store the password in a reversible (!encrypted) manner



The screenshot shows a Windows Command Prompt window titled "Administrator: C:\Windows\System32\cmd.exe". The command prompt displays the following text:

```
c:\tools>wce -w
WCE v1.42beta (X64) (Windows Credentials Editor) - (c) 2010-2013
- by Hernan Ochoa (hernan@ampliasecurity.com)
Use -h for help.

baltar\CYLONLOVER:Not even a Cylon could break this password.

c:\tools>
```

The number 46 is visible in the bottom right corner of the window.

WDigest FAIL

Above we see the use of Hernan Ochoa's Windows Credential Editor¹ (WCE), which was one of the first tools to expose the WDigest issue. With elevate privileges and running `C:\> wce -w`, we can see an extremely long cleartext password. That would be a fiendishly difficult to break hash, I expect.

[1] <http://www.ampliasecurity.com/research/wcefaq.html> (http://cyber.gd/511_352) QR



Microsoft Live Accounts: LiveSSP

- Microsoft is betting heavily on their cloud and subscription services
- Windows 8.1 and 10 work hard to convince you to login to Windows boxes with a Microsoft Account
- LiveSSP provides the integrated authentication for the Microsoft Account
 - Access to online storage via OneDrive, Office365, Outlook.com, etc.
- Imagine an adversary compromises one system where you use that account, could that expose the entirety of your online services?

Microsoft Live Accounts: LiveSSP

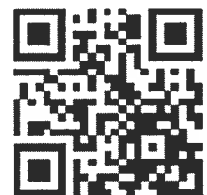
If you think WDigest is terrible and frightening, let's talk about LiveSSP¹. This recently added SSP supports Microsoft's initiative to get you a Microsoft account, purchasing their software as a subscription (Office 365), and get hooked on their cloud offerings (OneDrive being their personal cloud gateway drug of choice). A Microsoft Account, which used to be referred to as a Live account, is the single credential that unlocks all of Microsoft's public services.

As they would prefer you to begin consuming their subscription/cloud services, Microsoft wants to make the user experience as seamless as possible. To better enable this, Microsoft now includes the LiveSSP to extend the SSO capabilities.

However, imagine the potential damage of credential breach given that now not only is your internal corporate account compromised, but also the account used to access OneDrive, Office365, Outlook.com (Hotmail.com, Live.com), and more.

Unfortunately good documentation on LiveSSP is sorely lacking from Microsoft.

[1] <http://technet.microsoft.com/en-us/library/dd560645> (http://cyber.gd/511_353)



Microsoft Account FAIL

- Let's try to create a new Microsoft Account

16 character
maximum...

FAIL!!!

The screenshot shows the Microsoft account creation interface. At the top, there are two input fields for 'Name' with the values 'Karl' and 'Agathon'. Below this is a checkbox with the text 'After you sign up, we'll send you a message with a link to verify this user name.' followed by a 'User name' field containing 'helo@sec511.com'. Underneath is the option 'Or get a new email address'. A red error box highlights the password field with the message 'Your password can't be longer than 16 characters.' Below the error box is the 'Create password' section, which includes a password input field filled with dots and a note '8-character minimum; case sensitive'.

Continuous Monitoring and Security Operations

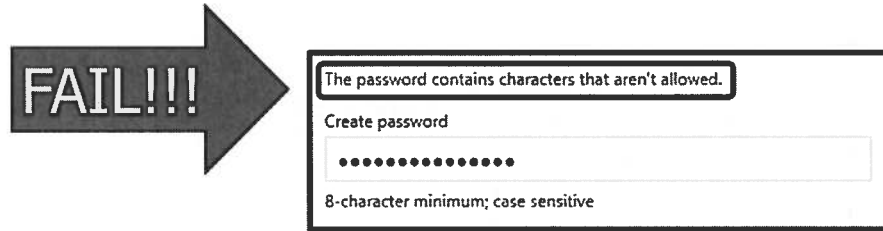
148

Microsoft Account FAIL

In the slide above we see an attempt to create a new Microsoft Account. Unfortunately the system currently only supports up to 16 character passwords. Given the recommendation to have ≥ 15 character passwords, we are fairly limited now in password length.

Let's Try That Again

- Well, at least if they are going to limit me to 16 characters for this critical account, then certainly they would support high complexity



Spaces also appear problematic

Continuous Monitoring and Security Operations

149

Let's Try That Again

At least if password length is going to be arbitrarily limited, then surely robust support for complex passwords would be available. As can be seen in the slide, this too appears to be lacking. Spaces appear problematic for Microsoft Account passwords.

LiveSSP FAIL

- Mimikatz from Benjamin Delpy (@gentilkiwi) exposes a much more serious issue

```
mimikatz # sekurlsa::livessp

Authentication Id : 0 ; 316271 (00000000:0004d36f)
Session          : Interactive from 2
User Name        : helo
Domain           : VIPERGOD
SID              : S-1-5-21-2013480549-2569447298-4169852308-1005

livessp :
* Username : helosec511@live.com
* Domain   : ps:password
* Password : Sharon!=Cylon
```

Up to 16 characters rendered in cleartext!

Continuous Monitoring and Security Operations

150

LiveSSP FAIL

The epic fail of Microsoft Accounts comes in the form of LiveSSP. The limited maximum length and complexity is bad, but what is vastly worse would be to go WDigest on us and actually store the password in a non-encrypted reversible form. The tool Mimikatz¹ from Benjamin Delpy (@gentilkiwi) shows that the password can be readily pulled from RAM on a system.

[1] <http://blog.gentilkiwi.com/mimikatz> (http://cyber.gd/511_354) **QR**



Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- **[Post-Authentication]**
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 151

Course Roadmap

Our next section is Post-Authentication.

Post-Authentication

- As will become clear during discussion of Access Tokens not all Logons are created equal
- There are various logon types tracked separately by Microsoft in the event logs
- Logging in at the console, over RDP, as a service, over the network, or with Cached Credentials each produce a unique Logon Type that can prove extremely helpful at identifying compromised credential abuse

Continuous Monitoring and Security Operations

152

Post-Authentication

Successful authentication with credentials yield a logon of some sort. There are different types of logons that bring with them different capabilities to the user. Also, some logon types are more likely to be targeted by adversaries as they are more powerful when abused.

Consider some of the various ways to logon to a Windows system:

- Console
- RDP
- As a service
- Over the network
- With cached credentials
- Etc.

Logon Types

- **Interactive Logon (Type 2)** – user logged on locally at the console
- **Network Logon (Type 3)** – authentication over the network
- **Service Logon (Type 4)** – account used to logon as a service
- **Unlock (Type 7)** – user account unlocked the workstation
- **Remote Interactive (Type 10)** – an interactive logon, like type 2, but over Remote Desktop Services
- **Cached Credentials (Type 11)** – authentication using cached credentials rather than the domain

Continuous Monitoring and Security Operations

153

Logon Types

Ways of logging in and the associated logon type¹ that would be referenced in the event logs.

Interactive Logon (Type 2) – user logged on locally at the console

Network Logon (Type 3) – authentication over the network

Service Logon (Type 4) – account used to logon as a service

Unlock (Type 7) – user account unlocked the workstation

Remote Interactive (Type 10) – an interactive logon, like type 2, but over Remote Desktop Services

Cached Credentials (Type 11) – authentication using cached credentials rather than the domain

[1] [http://technet.microsoft.com/en-us/library/cc787567\(v=ws.10\).aspx](http://technet.microsoft.com/en-us/library/cc787567(v=ws.10).aspx) (http://cyber.gd/511_321) QR



Access Tokens

- Once authenticated, Windows creates an Access Token for the user; this is the primary access token
- Copies of the Access Tokens are attached to each process and are used by the OS to determine what you are allowed to do
- In order to determine what you can do, the Access Token includes
 - User SID
 - Group Member SIDs
 - User Rights/Privileges
 - Integrity Label
 - Impersonation Level

Continuous Monitoring and Security Operations

154

Access Tokens

Beyond throwing a particular logon type in the Windows event log a more architectural change happens upon authentication. After authentication, users are provided with an Access Token¹. The access token will be attached to each process instantiated by the user in question. The access token is key to Single Sign On within Windows.

So what is actually included in the access token:

- User SID
- Group Member SIDs
- User Rights/Privileges
- Integrity Label
- Impersonation Level

[1] [http://msdn.microsoft.com/en-us/library/windows/desktop/aa374909\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa374909(v=vs.85).aspx)
(http://cyber.gd/511_315) QR



Token Impersonation Levels

- Besides Primary Access Tokens there are also impersonation tokens employed when a process acts like a user
 - Impersonation tokens are created with a set impersonation level
- Four impersonation levels
 - Anonymous
 - Identify
 - Impersonate
 - Delegate
- We are concerned most with impersonate and delegate

Continuous Monitoring and Security Operations

155

Token Impersonation Levels

Other than the Primary Access Tokens there are also impersonation tokens. These types of tokens are used by processes that are acting on behalf of the user. The primary access token defines an impersonation level, which will determine the capabilities associated the impersonation token.

There are four impersonation levels, but we are primarily only concerned with two types. The four levels are: anonymous; identify; impersonate; and delegate. The last two, impersonate and delegate are the ones we will focus on and have serious security implications.

[1] [http://msdn.microsoft.com/en-us/library/windows/desktop/aa378832\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa378832(v=vs.85).aspx)
(http://cyber.gd/511_355) QR



Impersonate Tokens

- The impersonation level of impersonate allows for systems to take actions as if they were us
- Though impersonate tokens are powerful and necessary, they do not allow for the impersonating process to interact as the initial user remotely
- The distinguishing feature of impersonate tokens is they are only used for local actions

Continuous Monitoring and Security Operations

156

Impersonate Tokens

Imagine that you are logged into a local Windows system and are accessing a remote service. What can you do on the remote system without actually logging in directly? This is where the concept of impersonation comes in. Understanding the impersonation level is critical for us.

Impersonate tokens allow for local interaction and impersonation of a security context. However, they specifically do not allow impersonation of the security context with respect to remote objects or resources. Impersonate tokens are associated with non-interactive logins.

Delegate Tokens

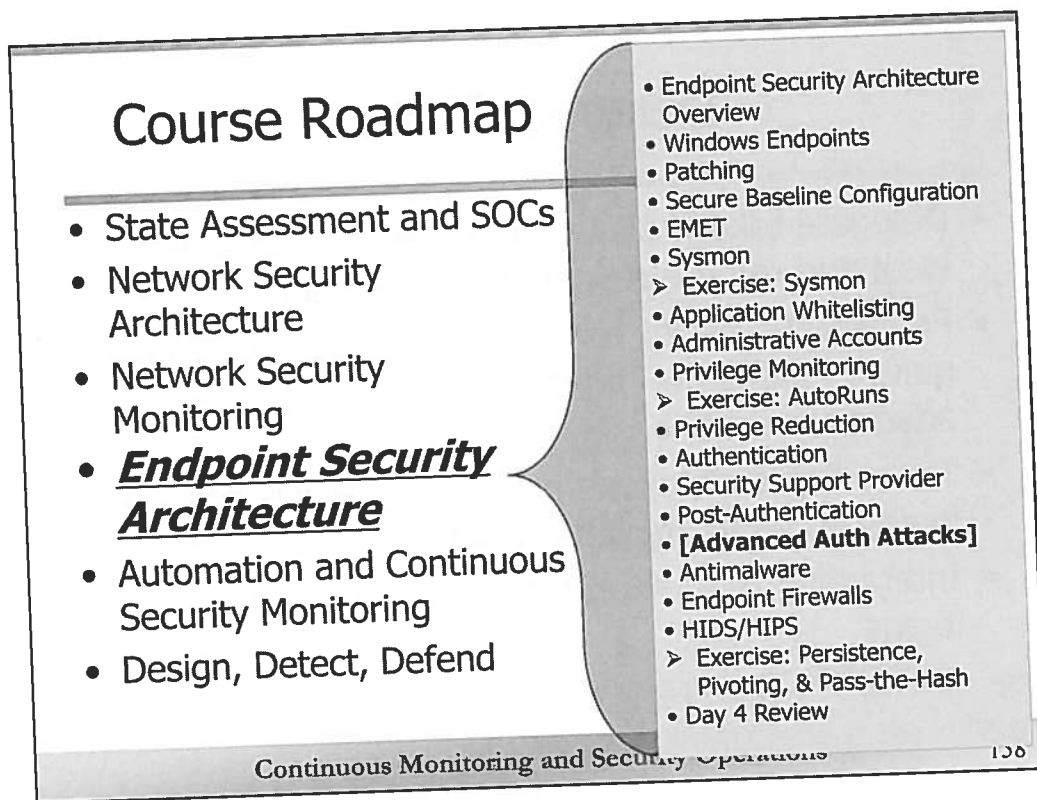
- Delegate tokens allow processes to access both local and remote resources as the user
- Privileged account tokens allowing delegation are quite possibly the highest value targets to an adversary
- These types of tokens are created when interactive logins (type 2) are performed
- Interactive login sounds like hands-on keyboard logins
 - RDP/VNC logins also constitute interactive logins

Delegation Tokens

Though impersonate tokens are important for security, delegate tokens are critical, and are very often abused by adversaries. The key distinguishing feature of delegate tokens is that they afford the ability to impersonate the security context even when accessing remote objects or resources.

Though often much less well understood than password hashes, access tokens in general, and delegate tokens specifically, very likely represent some of the highest value targets for adversaries.

Delegate tokens are created when interactive logins are performed. While hands-on-keyboard console logins are clearly interactive, RDP and VNC logins also constitute interactive logins.



Course Roadmap

Our next section is Advanced Authentication Attacks.

Pass-the-Hash

- One of the most common authentication attacks against Windows systems that targets local accounts and interactive logons
 - Leverages compromised hashes to remotely access other systems where the same username/hash exists
 - Lack of salts and synchronized accounts contribute to this attack's success
 - Commonly used to pivot in Windows shops where NTLM is still supported (read: almost all)
- More details (including mitigation steps) to follow in 511.5

Continuous Monitoring and Security Operations

159

Pass-the-Hash

Some of the most insidious attacks against Windows systems exploit architectural features rather than patchable flaws. Our advanced authentication attacks fit that bill. Pass-the-Hash (PTH) is probably one of the best known of these types of attacks.

The underlying basis of pass-the-hash is that NTLM network authentication starts with the hash rather than the password. The expectation is that if you can generate the hash then you must know the password to input to yield that hash. True, except when it's not (i.e. your hashes get compromised). The impact of this is significantly greater due to the fact that Windows does not employ salts for LM or NT hashes. This wouldn't be so bad if every account used different passwords, but usually there is at least one (administrator) that is commonly synchronized throughout the environment, or at least a chunk of it.

Traditional PTH is applicable to hashes available locally, which often means the primary victims are local accounts, but is often used in conjunction with the next attack technique to pivot deep within domains.

Alva "Skip" Duckwall (GSE # 40) and Chris Campbell have provided a lot of great information related to passing the hash on their joint blog¹.

Key tools for Pass-the-hash

- Windows Credential Editor
- Metasploit's psexec
- Mimikatz

[1] <http://passing-the-hash.blogspot.com/> (http://cyber.gd/511_357) **QR**



Token Smuggling - Pass the Session

- Rather than targeting local password hashes, this type of attack focuses on those access tokens
- In particular, the target is a delegate token
 - Preferably of a privileged user
- Stealing/reusing delegation tokens requires SYSTEM privileges on the endpoint where the token lives
- Token Smuggling is even more damaging than PTH because it is more widely applicable to domain accounts

Continuous Monitoring and Security Operations

161

Token Smuggling - Pass the Session

If you thought that PTH was bad, consider the next authentication attack token smuggling/pass-the-session. The primary reason this is a more significant issue is because it impacts domain accounts rather than simply locally available hashes. The first significant write up on this technique comes from Luke Jennings of MWR InfoSecurity¹.

The attack technique abuses the delegate tokens, typically of privileged domain users. By leveraging a delegate token from a privileged domain user, the adversary can possibly pivot even deeper into the enterprise than was possibly with PTH alone.

Key tools for token smuggling:

- Incognito
- Meterpreter's Incognito plugin
- Mimikatz

[1] https://labs.mwrinfosecurity.com/system/assets/142/original/mwri_security-implications-of-windows-access-tokens_2008-04-14.pdf (http://cyber.gd/511_356) QR



Pass the Pass(word)

- Benjamin Delpy (@gentilkiwi) has changed the face of authentication attacks multiple times with his great Mimikatz tool
- One of the most frightening series of blog posts have been his series "Pass the Pass(word)"
- Adversaries with **SeDebugPrivilege** or **SYSTEM** privileges can recover plain text passwords from several different SSP
 - TsPkg
 - WDigest
 - Live
 - **Kerberos**
- Adversaries might not need to bother with hashes/tokens

Continuous Monitoring and Security Operations

162

Pass the Pass(word)

The most egregious and frightening authentication attack has the adversary simply gaining our cleartext passwords directly. We saw a bit of this before with WDigest, but the issue is more pervasive. Benjamin Delpy (@gentilkiwi) and his tool Mimikatz have been fairly groundbreaking over the past few years on this front. Perhaps the most significant revelations have been put forth in his series of blog posts that started with an entry *Pass the Pass(word)*¹, in which Delpy illustrated that the TsPkg SSP, associated with Remote Desktop Services, exposed credentials in a way that allowed for recovery of cleartext passwords.

The next one to fall was the, now, infamous WDigest SSP that most of us didn't even realize existed². While WDigest was bad, it could be mitigated by disabling that SSP, but when LiveSSP fell next things were not looking so good for the home team. Perhaps the most shocking discovery though was of recoverable cleartext credentials were in the Kerberos⁴ SSP. Et tu, Kerberos.

[1] <http://blog.gentilkiwi.com/securite/pass-the-pass> (http://cyber.gd/511_358)

[2] <http://blog.gentilkiwi.com/securite/re-pass-the-pass> (http://cyber.gd/511_359)

[3] <http://blog.gentilkiwi.com/securite/rere-pass-the-pass> (http://cyber.gd/511_360)

[4] <http://blog.gentilkiwi.com/securite/rerere-pass-the-pass> (http://cyber.gd/511_361) QR



Mandiant M-Trends 2015 on Mimikatz

- Mandiant reports heavy attacker use of Mimikatz:
In nearly all of our investigations, the victims' anti-virus software failed to hinder Mimikatz, despite the tool's wide reach and reputation. Attackers typically modified and recompiled the source code to evade detection.¹
- Tools like Metasploit include some Mimikatz functionality
 - But the current native Mimikatz binary is typically more powerful and up-to-date
- How difficult is compiling a custom/altered version of Mimikatz?

Continuous Monitoring and Security Operations

163

Mandiant M-Trends 2015 on Mimikatz

Mandiant reports heavy use of Mimikatz in the cases they handled. Preventing and detecting use of Mimikatz on a network is an advanced, but critical mitigation.

Tools such as Metasploit do include Mimikatz functionality, but that functionality is typically limited and several generations behind the native Mimikatz binary, with Benjamin Delpy updates continuously. Therefore: after initial (local) system compromise, attackers will often attempt to copy an altered Mimikatz binary to the local file system run it, and use the stolen credentials to leverage domain access.

[1] <http://cyber.gd/m-trends-2015>

The Sed Persistent Threat (SPT)

- Windows mimikatz binary download
 - 70% AV detection rate
- Compiled mimikatz binary from source (**no** changes)
 - 31% AV detection rate
- Compiled **mimidogz** binary from source
 - s/mimikatz/mimidogz/g
 - 7% AV detection rate

virustotal	
SHA256	729962c639ac3b60cfeba01e5533ac814f078559164b257ecfb3a0c2bc94
File name:	mimikatz.exe
Detection ratio:	40 / 57
Analysis date:	2015-03-16 16:40:25 UTC (1 minute ago)
virustotal	
SHA256	1f577f342af40fc8c96c950ace9b2b9fcb2b9e53ef2a2d8ef5e9b31134733
File name:	mimikatz.exe
Detection ratio:	18 / 57
Analysis date:	2015-03-16 16:38:38 UTC (1 minute ago)
virustotal	
SHA256:	7066a2d24e102b6a776e99b040207aa1d502169512a0f8a5293a58d7b0311
File name:	mimidogz.exe
Detection ratio:	4 / 57
Analysis date:	2015-03-16 18:48:58 UTC (1 minute ago)

Continuous Monitoring and Security Operations

104

The Sed Persistent Threat

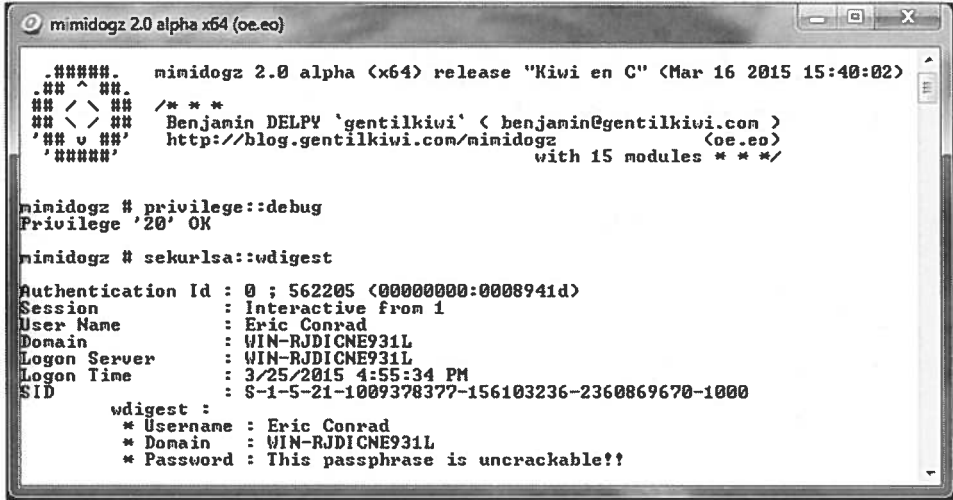
We jokingly call this approach the sed (stream editor) persistent threat as a knock on APT. Others use the term BPT (Basic Persistent Threat), or BAPT (Barely Adequate Persistent Threat).

There is a continuing myth that APT (Advanced Persistent Threat) is difficult to detect: in our experience this is simply not true. APT works well against cookie-cutter defenses that we have described in detail: the all-prevent defense, and a "set it and forget it" mentality.

In this case we decided to defeat 93% of antivirus vendors (including all the major vendors) by simply changed every Mimikatz source code file or directory with "mimikatz" in the name to "mimidogz", and we also made the same switch to the source file contents. Details of what we did are coming up next.

As you can see, this simple approach (it took <10 minutes to pull off the first time) was quite effective!

This Dog Can Hunt!



```
mimidogz 2.0 alpha x64 (oe.eo)

.#####. mimidogz 2.0 alpha (x64) release "Kiwi en C" <Mar 16 2015 15:40:02>
### ^ ###
### < > ### /* * *
### \ / ### Benjamin DELPY 'gentilkiwi' < benjamin@gentilkiwi.com >
'### u ### http://blog.gentilkiwi.com/mimidogz (oe.eo)
#####' with 15 modules * * */

mimidogz # privilege::debug
Privilege '20' OK

mimidogz # sekurlsa::wdigest

Authentication Id : 0 ; 562205 (00000000:0008941d)
Session : Interactive from 1
User Name : Eric Conrad
Domain : WIN-RJDICNE931L
Logon Server : WIN-RJDICNE931L
Logon Time : 3/25/2015 4:55:34 PM
SID : S-1-5-21-1009378377-156103236-2360869670-1000

wdigest :
* Username : Eric Conrad
* Domain : WIN-RJDICNE931L
* Password : This passphrase is uncrackable!!
```

This Dog Can Hunt!

As you can see, the mimidogz binary works perfectly, and is able to dump course author Eric Conrad's password: This passphrase is uncrackable!!

Note that the authors did not simply rename mimikatz to mimidogz: mimikatz is open source, so the authors changed every occurrence of the string or substring "mimikatz" to "mimidogz".

The commands shown above are:

```
# privilege::debug
```

This is required for Administrators. The system account does not require debug privilege to leverage this functionality.

```
# sekurlsa::wdigest
```

This dumps the wdigest passwords from RAM.

Whack-a-Mole

- We re-scanned mimidogz a few hours later on VirusTotal, and Kaspersky suddenly detected it
- We re-scanned the next morning, and 6 more vendors detected it (13 total)
- The total reached 26 vendors a week later

File name: mimidogz.exe	
Detection ratio	6 / 67
Analysis date	2015-03-18 22:56:58 UTC (0 minutes ago)
☒ Analysis ☐ File data ☐ Additional information ☐ Comments (0) Votes	
Antivirus	Result
Avast	Trusted Executable [Safe]
Avira	a variant of Win32/HackTool.Mimikatz
BitDefender	Trusted Executable [Safe]
ClamAV	Trusted Executable [Safe]
Comodo	Trusted Executable [Safe]
DrWeb	Trusted Executable [Safe]
Emsisoft	Trusted Executable [Safe]
Fortinet	Trusted Executable [Safe]
Ikarus	Trusted Executable [Safe]
Kaspersky	1333 (SuspiciousObject) Multi-Dropper

File name: mimidogz.exe	
Detection ratio	13 / 52
Analysis date	2015-03-19 12:12:12 UTC (1 minute ago)
☒ Analysis ☐ File data ☐ Additional information ☐ Comments (0) Votes	
Antivirus	Result
Avast	Trusted Executable [Safe]
Avira	Trusted Executable [Safe]
BitDefender	Trusted Executable [Safe]
ClamAV	Trusted Executable [Safe]
Comodo	Trusted Executable [Safe]
DrWeb	Trusted Executable [Safe]
Emsisoft	Trusted Executable [Safe]
Fortinet	Trusted Executable [Safe]
Ikarus	Trusted Executable [Safe]
Kaspersky	Trusted Executable [Safe]

File name: mimidogz.exe	
Detection ratio	26 / 55
Analysis date	2015-03-25 20:59:23 UTC (0 minutes ago)
☒ Analysis ☐ File data ☐ Additional information ☐ Comments	
Antivirus	Result
Avast	Trusted Executable [Safe]
Avira	Trusted Executable [Safe]
BitDefender	Trusted Executable [Safe]
ClamAV	Trusted Executable [Safe]
Comodo	Trusted Executable [Safe]
DrWeb	Trusted Executable [Safe]
Emsisoft	Trusted Executable [Safe]
Fortinet	Trusted Executable [Safe]
Ikarus	Trusted Executable [Safe]
Kaspersky	Trusted Executable [Safe]

Continuous Monitoring and Security Operations

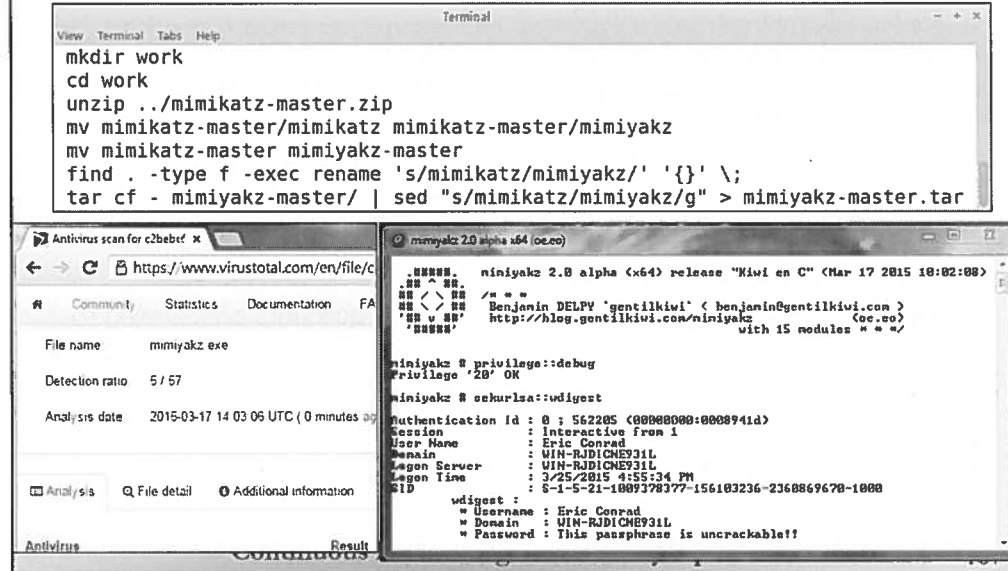
166

Whack-a-Mole

VirusTotal shares all samples with all vendors who participate. The evil, nefarious mimidogz was first picked up by Kaspersky. 25 other vendors eventually agreed that mimidogz was malicious.

Isn't that good? Well, not really. A real attacker will compile a custom mimikatz binary for each target. You can be sure the attacker will **not** upload that binary to VirusTotal.

Announcing MimiyaKz: The Sed Persistent Threat (SPT) Strikes Again!



Announcing MimiyaKz: The Sed Persistent Threat (SPT) Strikes Again!

To illustrate the futility of blacklisting: while antivirus vendors were busy blacklisting mimidogz, we made mimiyakz.

In the example above, we create a "work directory" and change to it, unzip the mimikatz source code to the work directory, rename two directories, recursively rename all files including the string "mimikatz" to "mimiyakz", and then tar the resulting contents, outputting to standard output.

The next step changes every "mimikatz" string and substring in the source code to "mimiyakz". The obvious (and inferior) way to do this is to open every file, search/replace, and then save the new files.

The superior (Unix "lazy") way is to use sed to perform a stream edit on the tar output. The tar (tape archive) command does not compress, unless you do so separately. So we can tar the contents to standard out, change every occurrence of "mimikatz" to "mimiyakz", and save a new tar file.

We then moved the tar file over to Windows, un-tarred it, and compiled it with Microsoft's Visual Studio Express Community 2013 (which is free):

<https://www.visualstudio.com/en-us/products/visual-studio-express-vs.aspx> (http://cyber.gd/511_380)

Advanced Authentication Attack Mitigations

- Most mitigations are partial fixes at best
- Many mitigations require Win8.1 and 2012R2
 - These remove most plaintext passwords from RAM by default, including wdigest
 - Livessp remain plaintext
- Application whitelisting to block/detect Mimikatz (and variants)
 - There are workarounds, but the attacker will likely trigger the whitelist first
- **Protected Users** - better Admin account protection
- **Restricted Admin Mode RDP** - no delegation token with RDP
- **Authentication Policy Silos** - control where accounts are allowed to be used
- Remove SeDebugPrivilege from accounts whenever possible
- Remove NTLM, if possible
- Require Smart Cards (at least for admins)

Continuous Monitoring and Security Operations

168

Advanced Authentication Attack Mitigations

Protected Users¹ - A new group available in AD with 2012R2, Protected Users, attempts to limit the impact of admin account compromise. Kerberos tickets expire sooner, no NTLM, no cached cred logins, no delegation tokens. On top of 2012R2, Windows 8.1 must be used for the admin systems.

Restricted Admin RDP is a way of remotely accessing systems via Remote Desktop Services without leaving a delegation token to be abused. Limited to Win8.1/2012R2. Launch with `C:> mstsc.exe /RestrictedAdmin`

Authentication Policy Silos - define areas in which credentials are allowed to be used. Requires 2012R2.

[1] <http://technet.microsoft.com/en-us/library/dn466518.aspx> (http://cyber.gd/511_365) QR

[2] <http://technet.microsoft.com/en-us/library/cc780217.aspx> (http://cyber.gd/511_366)

[3] <http://technet.microsoft.com/en-us/library/dn408190.aspx>
(http://cyber.gd/511_367)

[4] <http://technet.microsoft.com/en-us/library/dn486813.aspx>
(http://cyber.gd/511_368)



Two-Factor Authentication

- Password-only authentication has consistently been the most basic and popular approach to authentication
 - However, it has been shown to have a number of significant weaknesses as well
- Two-factor authentication requires an additional component beyond the password for interactive logon
- This could reduce some of the impact associated with third-party password breaches or direct hash compromise and cracking
- Two-factor authentication increases the strength and integrity of interactive logons
 - However, two factor is not the authentication panacea some believe

Continuous Monitoring and Security Operations

169

Two-Factor Authentication

Some industries and individuals seem to have an inflated perception of security when it comes to smart cards and two factor authentication. Especially with respect to these advanced Windows authentication attacks, the smart cards or other 2FA does not prove to be a tremendous stumbling block for adversaries.

That being said, for some scenarios 2FA provides significantly increased security over standard password-based authentication.

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- **[Antimalware]**
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations

170

Course Roadmap

Our next section is Antivirus and Antimalware.

Antivirus/Antimalware

- Just deploy it!
- Is antivirus alone sufficient? Of course not
 - It does catch widespread malware
- Yes, Antivirus/Antimalware has been getting a black eye for years
 - Main gripe is the signature-based detection component
- >20,000,000 malware specimens during a single quarter¹, courtesy McAfee
 - Enumerating all evil is always a losing proposition

Continuous Monitoring and Security Operations

171

Antivirus/Antimalware

Security professionals have long taken issue with basic antivirus/antimalware products. For years those professionals prone to say such things have declared antivirus to be “dead”. Unfortunately, some security practitioners have been listening carefully and perhaps might have been calling for the removal of AV. Interesting conversation, but not one I really care about. Just install it and go forth and get some other work done.

Breach is inevitable, I think we get that by this point of #SEC511. Do you want your company being the one testifying in a court and justifying removing the one security tool that most of the general public has a passing familiarity with?

McAfee during a recent report suggested that there were 20 million new malware specimens during one quarter of one year¹. Enumerating all badness and attempting to block it is a recipe for FAIL, and yet just keep deploying it and running it.

Look, I get it. AV is unlikely to be a hugely significant boon to your approach at catching evil. It is extremely far from perfect, and yet, just deploy it and keep moving.

[1] <http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q3-2013.pdf>
(http://cyber.gd/511_364) QR



More Support for AV

- An additional reason for supporting the deployment of AV has little to do with AV
- Most antivirus vendors have been feeling their customers' dissatisfaction
- Most commercial antivirus products are much more than just AV now
 - They are now fancy endpoint security suites
- The suites typically include many additional capabilities beyond pure AV
- Generally a dedicated point product will be more capable than the offering included within a general suite
 - But the standalone product will also usually be more expensive too

Continuous Monitoring and Security Operations

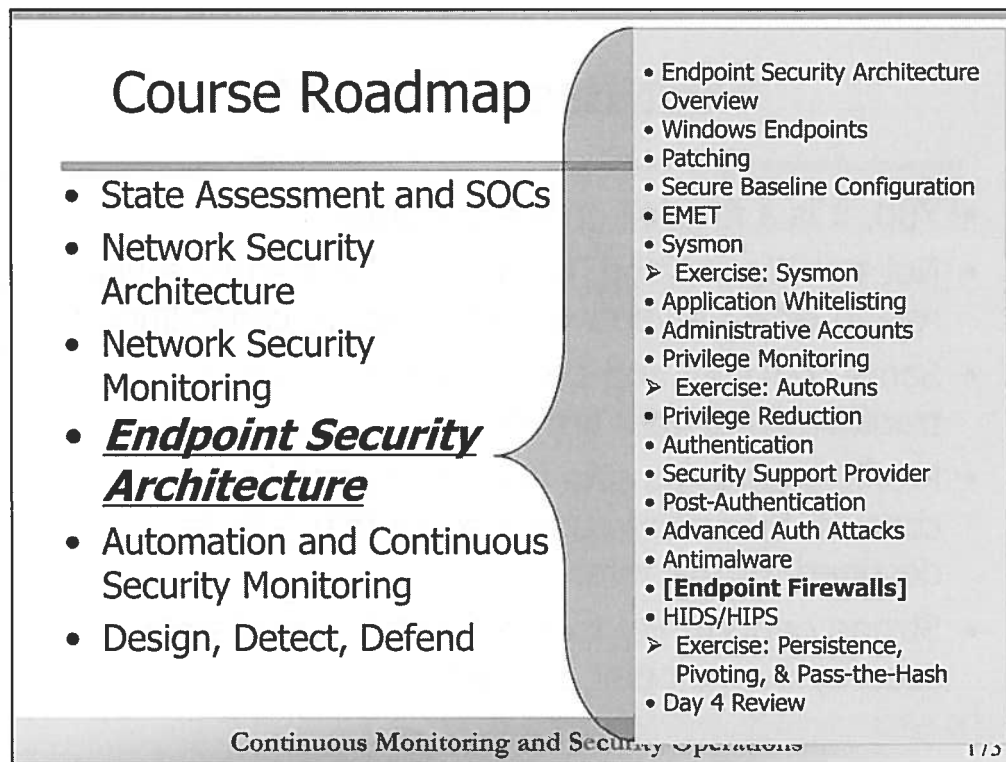
172

More Support for AV

Beyond just trying to scare you into continuing to spend capital and operational resources on AV, I do think there is actual positive security benefit. This benefit has precious little to do with antivirus itself. Our discontent with AV is palpable. Your vendors know you are dissatisfied with the offering, and once there is public precedent for a significant org shirking their antimalware responsibility then more will likely follow.

The play your antivirus companies have made is to rebrand themselves and their offerings to be much more than just antivirus. You typically do not have a subscription for AV software, but a rather more comprehensive endpoint security suite, or whatever your vendor decides to call it today.

These suites will be much more all-encompassing and will likely include standard, primarily signature based, antivirus, a host-based firewall, perhaps some HIPS capabilities, some HIDS capabilities though they are less overt about this one, perhaps too a dash of application whitelisting, removable device control, something they will suggest provides DLP (data leakage prevention capabilities, and more, and more, and more.



Course Roadmap

Our next section is Endpoint Firewalls.

Host-Based Firewall

- Yup, it is a firewall on the endpoint
- Not terribly exciting, but this tool can serve a vital role in both preventive and detective capabilities
- Same concepts and similar capabilities to a traditional network firewall
- Profiling egress traffic from endpoints is considerably more cumbersome than egress destined for the Internet
- Strong opportunity to greatly improve internal security in one point product

Continuous Monitoring and Security Operations

174

Host-Based Firewall

Day 2 explored the network firewall, but here we will attend to firewalling capabilities on the endpoint itself. Many of the same benefits of the more traditional network firewalls exist there.

Though the firewall is naturally and traditionally thought of as a preventive device, especially on the endpoints I find the more significant security boon comes from their logging capabilities.

Windows Firewall

- Talking about the firewall built into your modern Windows endpoints, technically Windows Firewall with Advanced Security (WFAS)
- Key aspects of WFAS
 - Free
 - Already installed
 - Managed via Group Policy
 - Network Location Aware
 - Egress Filtering
 - Stateful Packet Filtering
 - **Local** logging

Continuous Monitoring and Security Operations

175

Windows Firewall

Since the release of Windows XP SP2, and their fundamental shift towards substantially improved security, Microsoft has offered an endpoint firewall, Windows Firewall, as part of the basic OS. On modern Windows OS, the Windows Firewall has been rebranded as Windows Firewall with Advanced Security (WFAS).

So what does WFAS bring to the table? Most significant advantages are that it is FREE, already installed, and able to be natively administered via Group Policy. Other key features include network location awareness, which allows us to employ differing firewall policies for different networks we attach to, and egress filtering for controlling outbound traffic. WFAS represents a stateful packet filter firewall, which is to say that it actually keeps up with the state of connections rather than deciding permissibility based on each packet individually (stateless).

One pain point with WFAS is that the logs stay local by default. This is part of a larger historical pet peeve of mine with Microsoft and their lack of robust centralized logging (though we have seen marked improvements on this front recently).

Default WFAS

- Inbound filtered with a pre-configured list of allows
 - Most of the holes are for enabling Microsoft capabilities
 - You probably don't have much business need
- No egress filtering by default
 - Nontrivial to define a usable, but restrictive, configuration
- Logging disabled by default
 - Enabled: Still no built-in centralized logging
 - Enabled: 4MB total, not sufficient for detailed log
- No automated intrusion detection/monitoring capabilities without an external tool

Continuous Monitoring and Security Operations

176

Default WFAS

Though building a full endpoint firewall rulebase is beyond the scope of this course, we should at least have a conceptual understanding of the default WFAS configuration.

One of the most important considerations is that WFAS does not actually block any outbound traffic by default. Getting this configuration right for an organization can be cumbersome, but is a worthwhile initiative.

Another significant weakness, but one that is much easier to rectify is the poor logging configuration. By default logging allowed or even blocked connections is not enabled. These can easily be enabled. Another shortcoming is that the log limit is only 4MB. This likely wouldn't be a huge deal except that there is also no built-in functionality for centralized logging. Regardless, increase the log to its maximum of 32MB. Also consider, especially for laptops, configuring a separate log file for each profile. By default each profile uses the same log file which is in
`C:\Windows\System32\LogFiles\Firewall\pfirewall.log`

Not Windows Firewall

- Standalone desktop firewall deployments are rather rare
- Majority of organizations either
 - Use the Windows firewall
 - Use the firewall built into the deployed endpoint security suite
- Difficult to justify paying extra \$\$\$ for an endpoint firewall and then additional \$\$\$ for managing yet another agent

Continuous Monitoring and Security Operations

177

Not Windows Firewall

Though there used to be a market for standalone software-based commercial desktop firewalls, that time has largely gone away. If you have not tried you can imagine how fiendishly difficult it would be to cost-justify desktop firewall capabilities beyond what is already available for free.

The overwhelming majority of organizations will be leveraging either the Windows Firewall or perhaps a firewall product that is included within the overall endpoint protection/security suite you have already licensed.

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- **[HIDS/HIPS]**
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- Day 4 Review

Continuous Monitoring and Security Operations 1 / 6

Course Roadmap

Our next section is HIDS/HIPS.

Host-Based IDS/IPS

- One of the most common complaints about Network Intrusion Detection Systems is having to deal with the large volume of alerts generated
- Consider HIDS to be like miniature NIDS deployed on every endpoint
 - Now that is a lot of alerts to contend with
- Currently, the majority of organizations are more likely to employ HIPS than HIDS on endpoints
- Much like we saw with the endpoint firewall, it is relatively rare to see standalone HIDS/HIPS deployments
 - Again, as with the firewall, HIDS/HIPS is most often another piece of a larger endpoint security suite

Continuous Monitoring and Security Operations

179

Host-Based IDS/IPS

We discussed previously the notion that NIDS and NIPS were two entirely separate devices/controls (even if they just happen to actually be the same appliance). Though some of the same arguments can be made for the host-based IDS and IPS, the cost vs. benefits of maintaining both HIPS and HIDS will likely not be as supportive of maintaining both approaches.

What we are most likely to encounter will be a HIPS deployment that does provide some logging/alerting even HIDS-like functionality. Also, from a detective standpoint we often will be able to perform detection on the endpoints without a dedicated HIDS by performing log review/analysis, file integrity monitoring, etc.

HIPS are rarely seen as standalone software on endpoints any more, though they used to be somewhat common. Now, as with many other host-based security tools, the functionality has been, at least partially, rolled into the larger endpoint security suite.

Gains from HIDS/HIPS

- HIDS/HIPS can provide much needed internal visibility that is sorely lacking in most organizations
- After successful initial compromise, adversaries targeting your organization will inevitably pivot to target more important resources
- Even if we overlook the initial compromise, detecting the attempted, or even successful, pivot can mean the difference between a full blown data breach and a simple endpoint compromise

Continuous Monitoring and Security Operations

180

Gains from HIDS

One of the most significant justifications for HIDS/HIPS is that they are suitably positioned to provide a substantially improved degree of visibility within our enterprises. HIDS/HIPS do not require traffic to traverse a choke point where a gateway lives in order to provide value.

HIDS/HIPS are suitably positioned to better protect assets where key users, data, or applications reside. Perhaps the most significant WIN from HIDS/HIPS is their ability to help prevent, but most importantly detect attempts by adversaries to pivot. Even the approach discussed previously about employing VLAN ACLs to help with pivoting has a gaping hole of a blind spot when it comes to traffic staying on the same VLAN.

Detection without Response

- HIDS/HIPS more than almost any other tool generates a whole lot of data that could help identify compromise ourselves
- Of course throwing alerts without response doesn't provide much meaningful security benefit

BloombergBusinessweek Technology

Neiman Marcus Hackers Set Off 60,000 Alerts While Bagging Credit Card Data

By Ben Elgin, Dune Lawrence, and Michael Riley | February 21, 2014

These alerts were generated over several months, all of which were ultimately ignored

Continuous Monitoring and Security Operations

181

Detection without Response

The tremendous visibility into internal security can also prove problematic. The volume associated with HIDS/HIPS can be rather staggering for many organizations. This is especially true if they are attempting traditional detection where they wait for a tool to hit the big red evil button for them. Merely generating alerts is far from enough. The reason that we can so effectively peer into breaches after they occurred is because the data was there for the finding in the first place.

Just one of many examples of failure to proactively detect and respond to available data was in the case of the Neiman Marcus breach of 2013¹.

Detection must feed into response in order for it to be truly valuable.

[1] <http://www.businessweek.com/articles/2014-02-21/neiman-marcus-hackers-set-off-60-000-alerts-while-bagging-credit-card-data> (http://cyber.gd/511_37) **QR**



Host Detection without HIDS

- Keep in mind that tremendous detective capabilities are offered by host based security tools that are not HIDS
- Connections blocked by the endpoint firewall can be a significant detect
 - Highly useful for detecting attempts to pivot
 - Egress blocking can expose attempted C2
- Application Whitelisting blocks, after initial tuning, can also be a huge boon to detection
 - Adversaries want to persist on endpoints and typically will try to leverage an untrusted executable

Continuous Monitoring and Security Operations

182

Host Detection without HIDS

What if you don't have an overt HIDS tool, or even a HIPS tool that generates valuable data to mine for detects? Even if we did have a HIDS we would still want to leverage our other treasure troves of data. One of the significant ones that seems to get overlooked is the endpoint firewall. As we said previously, Microsoft, and most others too, do not enable connection logging by default for endpoint firewalls. Even if egress blocking never gets enabled just having those connections logs is of tremendous value. This is just one example of a detective capability without an intentionally detection-oriented tool.

Another significant source of valuable detects, the application whitelisting tool. Yes, it blocks unknown files from being executed, woohoo! I'd kinda like to know why something unknown was able to be there to be executed. How did it get there, is it malicious, any evidence of ultimately successful bypass? These are big important questions we need the answers to.

Approaches to HIDS

- System Integrity Monitoring (Baselining)
 - More robust configuration monitoring and tracking of security relevant configuration changes over time
 - See `snapshot.bat` from @JasonFossen
- File Integrity Monitoring
 - Simple critical file change monitoring
- Log Monitoring
 - Host-based log watcher that alerts if suspicious activity is recognized base on the event logs
 - See OSSEC from @danielcid

Continuous Monitoring and Security Operations

183

Approaches to HIDS

Some methods of doing overt Host Intrusion Detection include file integrity monitoring, system baselining, and log monitoring. These could be capabilities of a standalone tool, but often we can muster some of these capabilities on our own.

For example, Jason Fossen (@JasonFossen) provides a `snapshot.bat` file as part of scripts he wrote for #SEC505 and placed in the public domain¹. This simple script and the logs it creates can be leveraged to gain tremendous insight into adversary activities. His script is an example of system baselining, and can be a huge win for detection. File integrity monitoring is actually also instrumented into Jason's `snapshot.bat`

Another approach that needs to be instrumented is log monitoring. Spend a little time looking at high volume Windows Event Logs and you will scream until you can find a simple automated way to at least reduce some of the burden. Certainly any SIM/SIEM will provide this functionality, but we could also look at the outstanding open source tool OSSEC, created by Daniel Cid (@danielcid)².

[1] <http://cyber-defense.sans.org/blog/downloads/> (http://cyber.gd/511_362) QR

[2] <http://www.ossec.net/> (http://cyber.gd/511_363)



Post-Intrusion Detection

- Prefer to prevent compromise, but detecting a compromise while occurring would be a big win too
- Focusing on adversary post-exploitation activities proves successful
 - Persistence
 - C2
 - Pivoting
- Though suitably positioned, most traditional HIDS/HIPS tools lackluster for detecting these types of activities
- Additional tools, not classically considered HIDS/HIPS, can be a boon on this front that we will explore soon in an exercise

Continuous Monitoring and Security Operations

184

Post-Intrusion Detection

Though all would prefer to catch adversaries prior to compromise, detecting adversaries at any time before third-party notification of a breach would count as a win. For the majority of organizations this would also represent a significant change.

While we are starting to see some movement on this front, most of the security products are still overtly focused on preventing initial compromise rather than facilitating detection of eventual compromise. HIDS/HIPS offerings by virtue of where they are installed are suitably positioned to help us uncover key post-exploitation activities such as persistence, C2, and pivoting. However, many of them still seem rather poor on these fronts, nonetheless.

Some additional tools beyond classic HIDS/HIPS could potentially help fill this gap for us.

Memory Analysis

- “**Malware can hide, but it must run**,” tagline for SANS Memory Forensics (FOR526) class¹
- To achieve persistence, C2, or pivoting malware must execute, and will, by necessity, show up in running memory
- Dumping and analyzing memory from a system can expose adversary behaviors otherwise obscured or seemingly innocuous
- Memory analysis can expose
 - Running processes
 - Network connection details
 - Injected code (DLL injection)
 - Much more
- Full memory captures can allow for extraction of binaries executed even if they no longer exist on the hard disk
 - Allows subjecting code to threat intelligence, dynamic, or static analysis

Continuous Monitoring and Security Operations

185

Memory Analysis

In the forensics world there has been a significant surge in emphasis on memory forensics and analysis. As malware and adversaries continue to get more sophisticated they will, no doubt, increasingly attempt to remain undetected. While in many organizations this seems to pose little problem, we have seen modern malware that includes sophisticated techniques to avoid even advanced prevention and detection capabilities. The tagline of SANS Memory Forensics class (FOR526) is “malware can hide, but it must run.”¹ Any time malware executes, there will necessarily be traces in memory regardless of how stealthy the adversary attempts to be. Naturally, to persist, communicate over C2, or pivot, the malware must run.

Memory analysis capabilities have increased dramatically over the years. Though there are even several well maintained free memory analysis tools, this space requires constant updates to be able to analyze captures from the ever-changing memory artifacts associated by systems and applications. Some key artifacts available for discovery via memory analysis include running processes, injected code such as injected DLLs, network connections, passwords, and much more. Further, if a full memory capture exists, then the analyst can potentially carve out suspect binaries. These binaries can then be submitted to public threat intelligence sources. Further, dynamic or static analysis of the code can be performed using automated tools, or by hand, with skilled analysts.

[1] <http://www.sans.org/course/memory-forensics-in-depth> (http://cyber.gd/511_378)

Redline

- Free and easy to use memory analysis tool from Mandiant/FireEye
 - Integration with IOCs to ease looking for known campaigns
- Includes a small hash whitelist of known good
 - Larger set available for download
 - Supports addition of custom whitelist hashes
- Timelines are a key focus of Redline reports
- Provides a GUI for reviewing the details of the analysis
 - Mandiant's open source Python-based AuditParser can be used to convert the binary .mans files to tab-delimited text files for command line parsing
- Though primarily considered a memory analysis tool, Redline also reports additional registry and file-based information

Continuous Monitoring and Security Operations

186

Redline

First and foremost, Redline is considered a memory analysis tool¹. Under the hood, Mandiant's Memoryze is used for the acquisition of a memory image. Many alternatives exist for performing memory analysis, and especially acquisition. Redline, however, is consistently updated, free, and easy to use. The last element especially can be a boon for analysts. Memory forensics/analysis is an advanced capability that requires significant expertise to perform well. Redline lowers the bar considerably to allow for those less well-versed in the details of memory analysis, to still achieve benefits.

Beyond the basic memory analysis capabilities already discussed, some features of Redline warrant additional discussion. Redline allows for the analyst to supply a file that contains hashes of known good files to be included to reduce the likelihood of false positives in the Redline output. Custom hashes can be supplied, but Redline comes with a limited set of hashes built-in, and also allows for downloading a file that includes many more². For those familiar with Mandiant, it will come as no surprise that Redline can leverage IOCs to help guide the analysis. Obviously, this is particularly useful if part of your organization's standard security operations includes creating IOCs for incidents or malicious campaigns.

From an analytic standpoint, Redline emphasizes a timeline-based approach to investigation, which is commonly associated with forensics and incident response. This approach can help identify the cause of changes being experienced or logged elsewhere.

The Redline GUI is typically used for reviewing the results, but Mandiant has made available an open source Python-based tool called AuditParser to convert the .mans binary file format to a collection of simple text files that can be parsed from the command line³.

[1] <https://www.mandiant.com/resources/download/redline> (http://cyber.gd/511_376) **QR**

[2] Ibid.

[3] <https://github.com/mandiant/AuditParser> (http://cyber.gd/511_377)



Kansa - Go Big/Wide or Go Home

- Memory analysis can be an incredible way to discover even the more subtle adversaries
 - However, analyzing memory of individual systems scales poorly
- Kansa by Dave Hull (@davehull)
 - Open source PowerShell based IR framework
 - Collectors use PS Remoting to pull from many systems at once
 - Analysis scripts can highlight items of interest in resultant large datasets, which proves incredibly effective for hunt teaming
 - Leverages what Dave refers to as stack ranking (long tail analysis in SEC511) building upon his Get-StakRank PS script
- Kansa pay dirt, finding the outliers across many Windows systems (E.g. uncommon services, listening ports, processes, DNS cache entries)

Continuous Monitoring and Security Operations

188

Kansa - Go Big/Wide or Go Home

One of the key challenges associated with memory analysis is scalability. Memory acquisition is time consuming and, for full captures, is a significant volume of data. That is just on the acquisition side. Analyzing the acquired memory image is also time consuming. Scaling this across an enterprise becomes very challenging and/or expensive. To scale memory analysis typically implies enterprise forensic/IR software be installed on each endpoint as an agent, and still would typically require ad-hoc acquisitions to be performed. For these reasons, memory analysis is typically performed after there is sufficient reason to warrant this level of investigation. While intentional detailed analyses will still be performed, how might we gain detailed intelligence at scale to help find those items that our blacklisting-oriented tools fail to uncover?

An open source project from Dave Hull, Kansa, provides an interesting potential solution to this problem of IR style intelligence at scale. Kansa is a PowerShell based IR framework¹. PowerShell remoting allows for being able to execute Kansa against remote Windows systems with sufficient privileges to capture detailed information key to many investigations. It also allows for pulling this information from a large number of systems. Much additional information about both usage and use cases of Kansa can be found at Dave's blog². Though the tool is presented as being focused on IR, the capabilities can be leveraged for significant gain on the hunt team front. Later additions to the tool really honed the hunt team aspects of the tool by including analysis scripts that parse the results of the collectors in meaningful ways.

Dave leverages what he refers to as stack ranking, which we call long tail analysis in SEC511, to mine collected data for potentially actionable items based on frequency analysis. The analysis scripts build upon his Get-StakRank PowerShell script to perform frequency analysis against delimited text file output³. The classic example of using this analytic approach to lead to a significant finding is through considering Windows services.

Adversaries want to persist. A common means of doing so is through the creation (or even co-opting) of a service. Services are normal. Creation of new services can be normal and legitimate. Investigating every service creation or change to an existing service, while important, might be easily overlooked, and would frequently mean analyzing legitimate services. Stack ranking, or performing long tail analysis, of services can quickly yield items warranting investigation by highlighting those infrequently occurring service names or even a combination of service names, path to executable, and even hash of executable. Outliers that look innocuous by having an expected file name or reusing a common service name, would easily surface under this type of scrutiny. This and many similar analyses can be performed rapidly, and repeatedly, by leveraging Kansa.

[1] <https://github.com/davehull/Kansa> (http://cyber.gd/511_373) **QR**

[2] <http://trustedsignal.blogspot.com/search/label/Kansa> (http://cyber.gd/511_374)

[3] <https://github.com/davehull/Get-StakRank> (http://cyber.gd/511_375)



Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - **[Exercise: Persistence, Pivoting, & Pass-the-Hash]**
- Day 4 Review

Continuous Monitoring and Security Operations 190

Course Roadmap

Our next section is the Persistence, Pivoting, and Pass-the-Hash exercise.



SEC511 Workbook Persistence/Pivoting/Pass-the-Hash

Exercise 3: Persistence, Pivoting, and Pass-the-Hash

Continuous Monitoring and Security Operations

191

SEC511 Workbook Persistence/Pivoting/Pass-the-Hash

Please go to the 511 Exercise Workbook, section 511.4-3.

Course Roadmap

- State Assessment and SOCs
- Network Security Architecture
- Network Security Monitoring
- **Endpoint Security Architecture**
- Automation and Continuous Security Monitoring
- Design, Detect, Defend

- Endpoint Security Architecture Overview
- Windows Endpoints
- Patching
- Secure Baseline Configuration
- EMET
- Sysmon
 - Exercise: Sysmon
- Application Whitelisting
- Administrative Accounts
- Privilege Monitoring
 - Exercise: AutoRuns
- Privilege Reduction
- Authentication
- Security Support Provider
- Post-Authentication
- Advanced Auth Attacks
- Antimalware
- Endpoint Firewalls
- HIDS/HIPS
 - Exercise: Persistence, Pivoting, & Pass-the-Hash
- **[Day 4 Review]**

Continuous Monitoring and Security Operations

192

Course Roadmap

Our next section is an exercise with OSSEC.

Day 4: Punch List/Action Items

- Deploy Application Whitelisting
 - Only allow previously identified and vetted binaries to execute
- Remove key Windows privileges
 - Most importantly, remove Debug Programs privilege from all user accounts that lack explicit need
- Disable the built-in administrator account
 - Review any/all attempts to interact with this account
- Review and revoke excessive user rights
 - Target servers/services accounts to block local logon

Day 4: Punch List/Action Items

The punch list/action items is your homework. What are some key takeaways for you to immediately go back to your organization and affect change? Your instructor has, no doubt, also provided some additional items to be included in your punch list, but this slide provides a quick sanity check refresh of some key actions for you to make sure to hit upon return to your workplace.

Day 4: TL;DR

- CSC First Five Quick Wins
 - Does your organization meet those requirements?
- Application Whitelisting must become de facto for all organizations concerned with security
- Abuse of authentication credentials is rampant in compromise
 - Plan accordingly
- Windows permissions, privileges, and rights play a significant role in internal security
 - And we know adversaries become or abuse insiders...

Continuous Monitoring and Security Operations

194

Day 4: TL;DR

TL;DR is a common shorthand for Too Long; Didn't Read and is often put at the top of long e-mails or blog postings that go into tremendous detail. For our purposes this is a quick high level summary of major ideas/themes from the day's material.



Immersive Cyber Challenges



Continuous Monitoring and Security Operations

195

SEC511 Daily NetWars

Connect to the daily NetWars environment and continue working through the SEC511: Immersive Cyber Challenges.

Please see the first lab (511.1-0) in the SEC511 Workbook for details and instructions on configuring your system to connect to the NetWars environment.

Note: As indicated by the icon above, this lab leverages the class network. OnDemand, vLive, Simulcast, or other online students will need to connect to the SEC511A VPN to complete this lab.

