504.1

Incident Handling Step-by-Step and Computer Crime Investigation

SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

504.1

Incident Handling Step-by-Step and Computer Crime Investigation

SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

Copyright © 2016, The SANS Institute. All rights reserved. The entire contents of this publication are the property of the SANS Institute.

PLEASE READ THE TERMS AND CONDITIONS OF THIS COURSEWARE LICENSE AGREEMENT ("CLA") CAREFULLY BEFORE USING ANY OF THE COURSEWARE ASSOCIATED WITH THE SANS COURSE. THIS IS A LEGAL AND ENFORCEABLE CONTRACT BETWEEN YOU (THE "USER") AND THE SANS INSTITUTE FOR THE COURSEWARE. YOU AGREE THAT THIS AGREEMENT IS ENFORCEABLE LIKE ANY WRITTEN NEGOTIATED AGREEMENT SIGNED BY YOU.

With the CLA, the SANS Institute hereby grants User a personal, non-exclusive license to use the Courseware subject to the terms of this agreement. Courseware includes all printed materials, including course books and lab workbooks, as well as any digital or other media, virtual machines, and/or data sets distributed by the SANS Institute to the User for use in the SANS class associated with the 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.

BY ACCEPTING THIS COURSEWARE YOU AGREE TO BE BOUND BY THE TERMS OF THIS CLA. BY ACCEPTING THIS SOFTWARE, YOU AGREE THAT ANY BREACH OF THE TERMS OF THIS CLA MAY CAUSE IRREPARABLE HARM AND SIGNIFICANT INJURY TO THE SANS INSTITUTE, AND THAT THE SANS INSTITUTE MAY ENFORCE THESE PROVISIONS BY INJUNCTION (WITHOUT THE NECESSITY OF POSTING BOND), SPECIFIC PERFORMANCE, OR OTHER EQUITABLE RELIEF.

If you do not agree, you may return the Courseware to the SANS Institute for a full refund, if applicable.

User may not copy, reproduce, re-publish, distribute, display, modify or create derivative works based upon all or any portion of the Courseware, in any medium whether printed, electronic or otherwise, for any purpose, without the express prior 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.

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.

SANS acknowledges that any and all software and/or tools, graphics, images, tables, charts or graphs presented in this courseware are the sole property of their respective trademark/registered/copyright owners, including:

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.

Governing Law: This Agreement shall be governed by the laws of the State of Maryland, USA.

SEC504.1 | Hacker Tools, Techniques, Exploits and Incident Handling

SANS

Incident Handling Step-by-Step and Computer Crime Investigation

Copyright 2016, Ed Skoudis, John Strand, SANS | All Rights Reserved | Version B01_02

Hello and welcome to SANS Security 504, "Incident Handling and Hacker Exploits." We will spend the next several days discussing how attackers break into systems, and, more importantly, how you can prevent, detect, and respond to such activities. This material is designed to help prepare you for the GIAC GCIH certification.

Our initial focus is on incident handling as we discuss time-tested procedures for responding to computer attacks. The incident-handling approach you will learn was originally developed by the United States Department of Energy and then adopted by the U.S. Navy. Since then, this process has been further developed and refined by hundreds of incident handlers who have worked for over a decade to improve the state of practice. These efforts have centered on making the approach more generalized to support corporations, government agencies, educational institutions, and other organizations. The focus of this class is to prepare you to handle an incident.

SANS is serious about this training and certification, and we have invested a lot of time and effort into building a class that will prepare you to handle just about anything! You are going to have to work to win the certification, but you will know that it truly means something once you've achieved it. Incident handling is not a stand-alone skill; it builds on your system administration and network defense training. When you are under fire, you will appreciate having these skills at the ready!

We would be delighted if you would join our mailing list for 504 news and updates. You can find it at:

http://eepurl.com/ZhFfn

TABLE OF CONTENTS

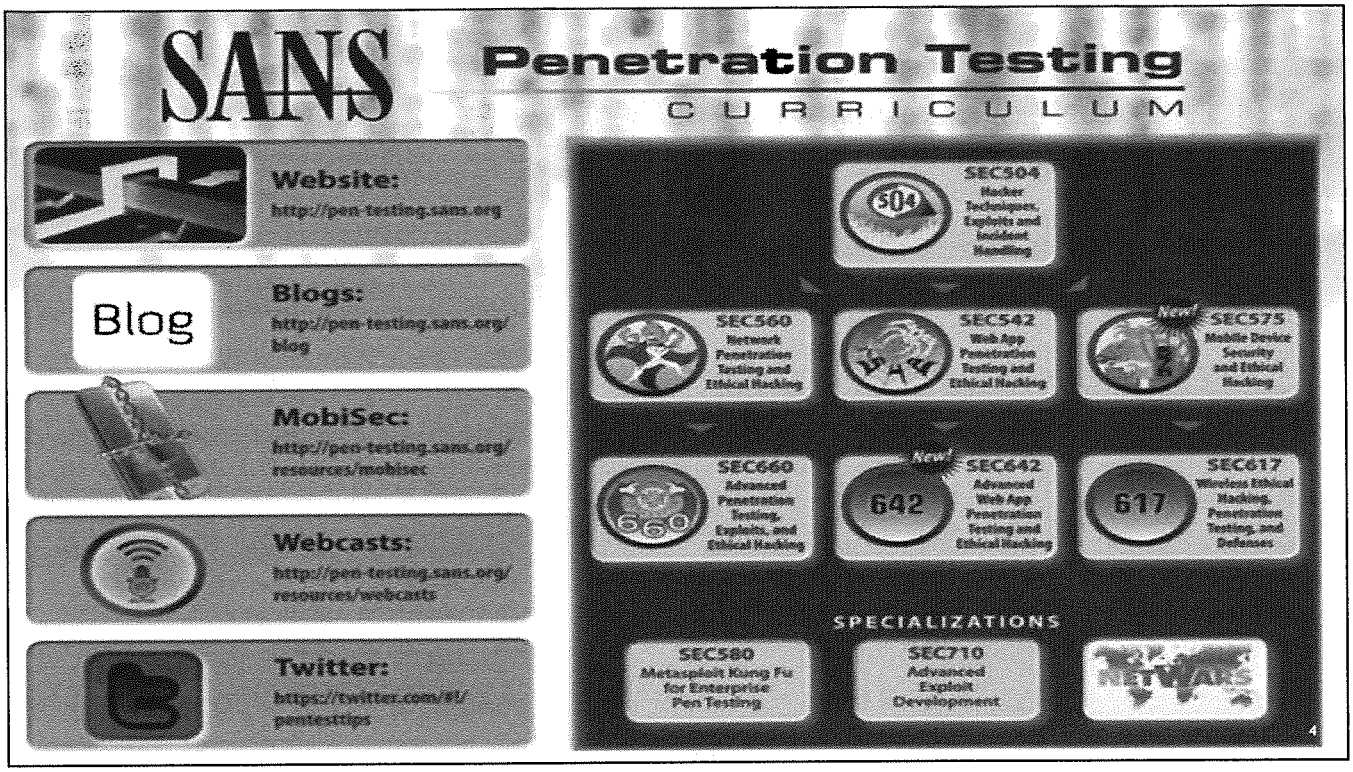
This page intentionally left blank.

© 2016 Ed Skoudis, John Strand

TABLE OF CONTENTS

This page intentionally left blank.

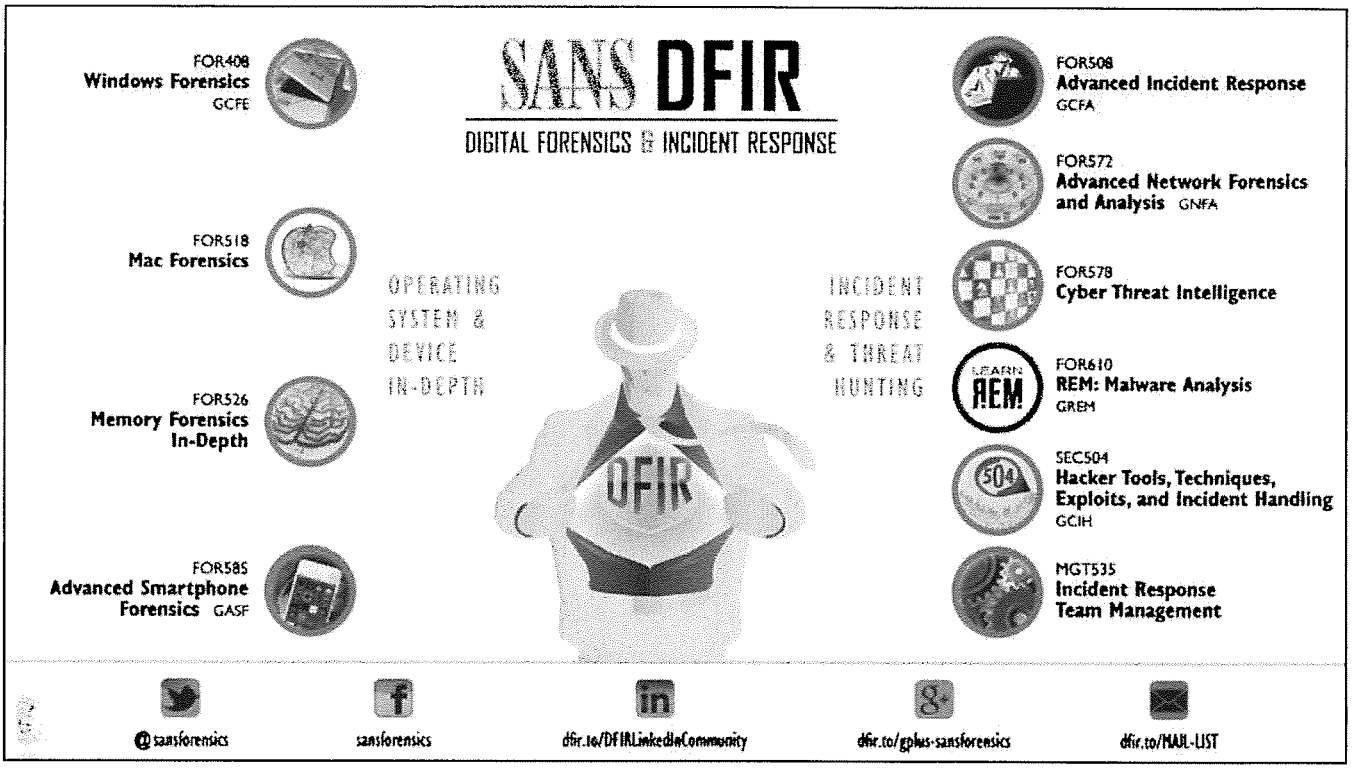
© 2016 Ed Skoudis, John Strand



The SANS 504 course covers a variety of attacks and associated defenses. It explains how you can apply incident-handling procedures to address each step of an attack. It is built around the philosophy that offense must inform defense. That is, to be a solid defender, you need to understand the attacks your systems and networks will face every day. Along with that notion is the concept that to be a good attacker (such as a penetration tester or red teamer), you need to know the defenses. There are two important reasons that professional attackers need to understand defenses. First, penetration testers need to be able to make recommendations about what kinds of defenses should be in place in your report and recommendations. Furthermore, penetration testers need to understand defenses because they need to consider ways to thwart or bypass them.

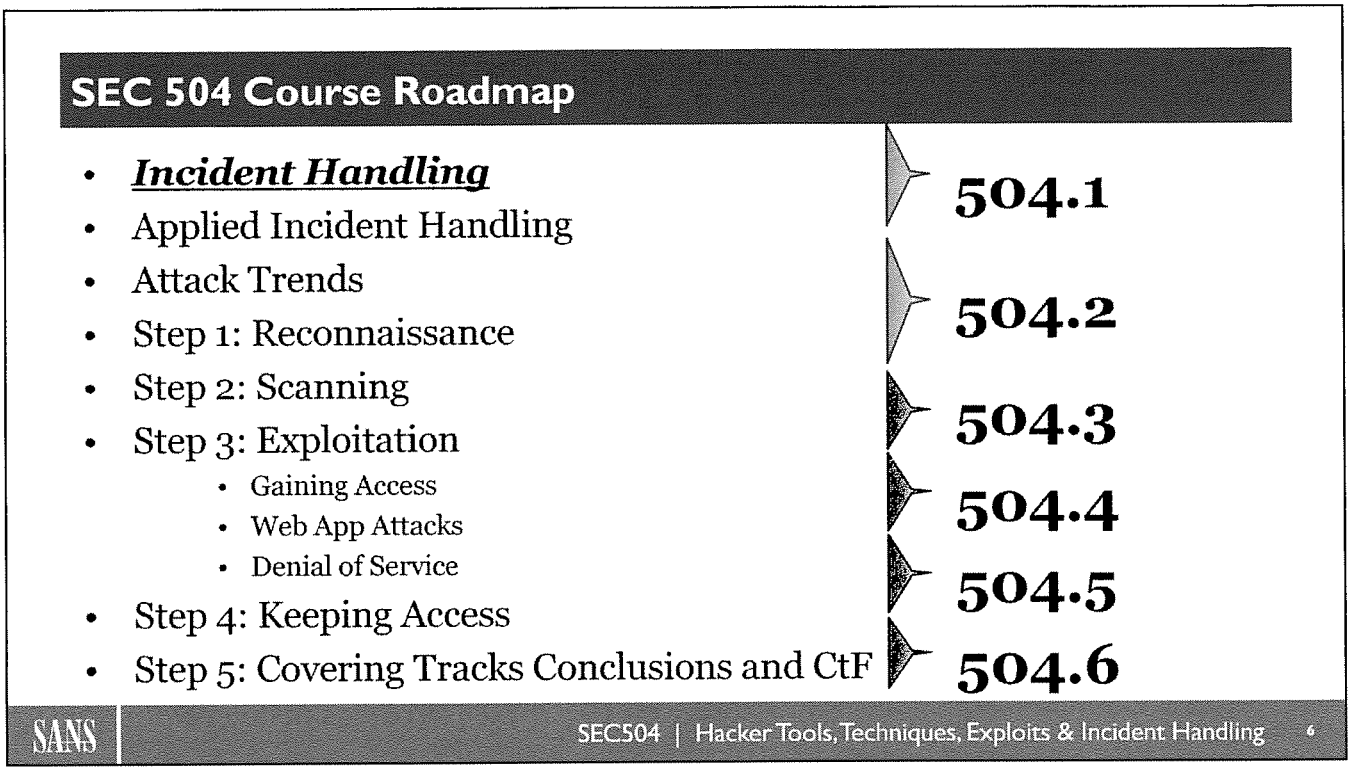
For those reasons, this SANS 504 course is in a crucial location in the SANS Penetration Testing Curriculum and the SANS Digital Forensics and Incident Response (DFIR) Curriculum. In the slide, you can see its location in the Penetration Testing Curriculum, covering insights into a variety of attacks (and their associated defenses) and providing security professionals detailed foundations and capabilities for understanding, analyzing, and launching attacks and applying practical defenses.

© 2016 Ed Skoudis, John Strand



As described on the previous slide, defenders need a solid understanding of attacks. When offense informs defense, security personnel, such as digital forensics experts, can anticipate an attacker's moves and analyze or counter them much more effectively. For that reason, this 504 course is also in a critical position in the SANS DFIR curriculum, as you can see in the slide.

© 2016 Ed Skoudis, John Strand



This course is made up of six parts. In the first part, 504.1, we discuss incident-handling techniques, focusing on a well-established process for handling incidents in enterprises. We discuss the overall methodology, picking up useful processes and technical tips along the way. In addition, we go over computer crime topics and the legal system, discussing how they impact incident handlers.

After establishing a firm base of incident-handling concepts, the course shifts into a discussion of how attackers exploit target systems and networks, from a step-by-step perspective. We go over reconnaissance and scanning in 504.2 to see how attackers get to know a target environment better. Next come two of the biggest components of the course, 504.3 and 504.4, in which we discuss how attackers gain access to target machines. This section is split over two course books because attackers have so many methods to infiltrate targets.

Section 504.5 looks at the next phases of many attacks after attackers gain access: maintaining access and covering their tracks. We address topics such as backdoors, rootkits, and log editing.

Our final component of the course is designed to help get attendees into the mindset of attackers. This gives you an idea of how an attacker views a target environment and the steps he or she would apply. This last topic, covered in 504.6, is our Capture the Flag (CtF) event, in which you get to apply all of the attack phases into a competition to hammer home lessons from throughout the course.

© 2016 Ed Skoudis, John Strand

504 VM Setup

- The 504 Class VM is an .ovf
  - This is an open virtual machine format
  - Simply double-clicking on the .ovf file should start the import process
  - If you get .ovf consistency errors, check just select "Retry"
- However, if the import process does not start, simply select File > Import in VMware
- Start this process now, as it can take some time
- All passwords and some usage hints are in the Release Notes files
- Set your IP address when you log on with # `ifconfig eth0 10.10.75.1/16`
- The passwords for the VM are:
  - sec504 = sec504
  - root = $sudo su – (then type "sec504" as the password)
- If you need to change your keyboard layout, simply select the following after logging in:
  - Activities > Type "keyboard" > Then choose "Region & Language"

In this class, we use the .ovf format for the course VM. We do this because it is compressed and can be ported very easily to a wide variety of different virtualization platforms.

In most cases, simply selecting the .ovf file will start the import process. However, if it does not work, simply open VMware and select File > Import and navigate to the .ovf file. On some versions of VMware, you simply choose "Open a Virtual Machine," and then navigate to the .ovf file. You might get a consistency error from VMware. This is because the OVF was not created with the same version of VMware you have. Just select Retry and it should import just fine.

We recommend you start this process now as it can take some time to import this VM.

Finally, a couple of additional notes. First, the passwords for the VM are in the Release Notes. We have provided the Release Notes in .txt and .doc format.

Next, please do not use this VM for any production-related activities. It should be used only for this class. It is not regularly updated or secured. The passwords for the VM are in the release notes for the class. But for immediate reference, they are also listed here:

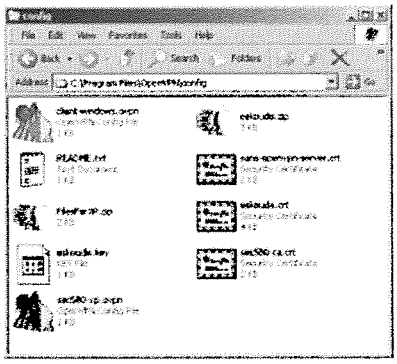
sec504 = **sec504**
root = **$ sudo su – (The type s"sec504")**

Finally, for our international students, welcome. If you need to change your keyboard layout, please select the following:
Activities > Type "keyboard" > Then choose "Region & Language."

© 2016 Ed Skoudis, John Strand

VPN Configuration for vLive and OnDemand

- If you are taking this class across the Internet (either via SANS vLive or SANS OnDemand), you will receive an e-mail with instructions for getting networked across the VPN
- The e-mail will explain how to:
  - Network your host and guest machines on the Internet; make sure both can access the web by pinging www.google.com
  - Download the OpenVPN install files for Windows and your certificates
  - Install OpenVPN on Windows, and place your certs in the appropriate place
  - On Linux, download and place your certificates in the appropriate place
  - Establish VPN connection from Windows
  - Establish VPN connection from Linux
  - Make sure both can ping 10.10.10.45



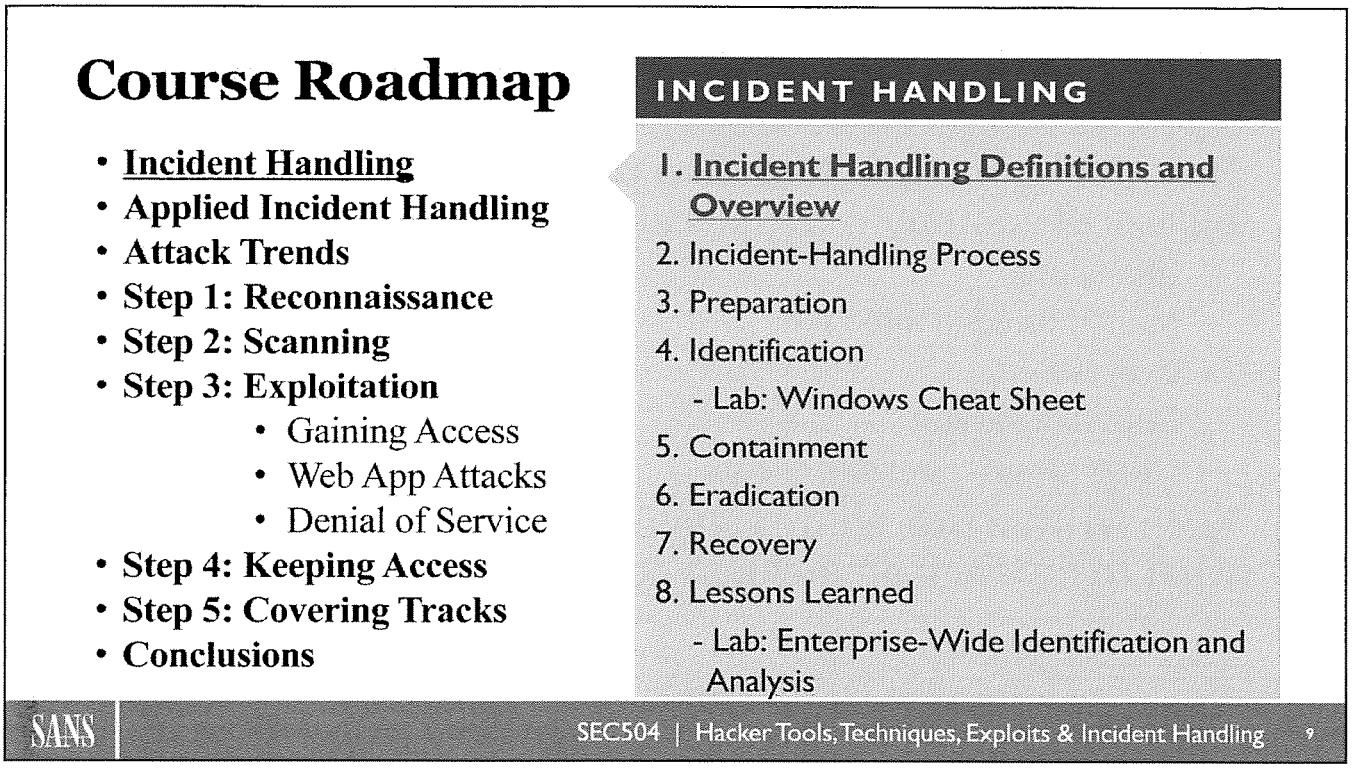
If you are taking this course across the Internet (either via SANS vLive or SANS OnDemand), you need to set up OpenVPN on your Linux and Windows machines to conduct the bridged networking labs in the class so that you can reach target systems we have prepared.

You will receive an e-mail from SANS NOC personnel that describes in detail the process for configuring your system to use the VPN. The e-mail will explain various steps, including

1. Set up your Linux guest and Windows host machine on the Internet. Both machines must reach Internet destinations. For Linux, use bridged networking, and configure your guest machine with an IP address for your environment or pull one using DHCP (edit /etc/network/interfaces). If you use hard-coded IP addresses, simply set it in the line that shows the IP address for eth0. If you use DHCP, make sure you change static to dhcp. Make sure both your Windows and Linux machines can ping some site on the Internet, such as www.google.com.
2. Download the OpenVPN install files for Windows, along with your certificates, as described in the e-mail from the SANS NOC. Put your certificates in the appropriate place (C:\Program Files\OpenVPN\config). You do not need to install OpenVPN software on the Linux guest image we provided because this software is already installed.
3. On Linux, place your downloaded certificates in the appropriate place (/etc/openvpn).
4. Establish the VPN connection from Windows (by right-clicking the OpenVPN icon in your tooltray and selecting Connect). Provide your SANS portal password to connect.
5. Establish the VPN connection from Linux (by running service openvpn start). Again, provide your SANS portal password when prompted.
6. Make sure both Windows and Linux can ping 10.10.10.45 while the VPN is connected.

Note: To communicate between your Linux guest and Windows host while connected to the VPN, you could use the IP address assigned to you by the VPN (viewable via the OpenVPN tooltray client in Windows and as the tap0 interface displayed by the ifconfig command on Linux) or the IP address of your network adapter (Local Area Connection in Windows and eth0 in Linux).

© 2016 Ed Skoudis, John Strand



The first day's materials are broken into the following phases:

- **Overview**: This is an introduction to the material.
- **Detailed Incident Handling**: This is the heart of the 504.1 materials. It describes a six-step process for handling incidents in depth.
- **Incident Tips**: This section presents tips and tricks for various scenarios. Each suggestion has served experienced incident handlers well.
- **Law, Crime, and Evidence**: In this component, we address the various laws that an incident handler may encounter.

We also include some additional material in the appendices that you'll need to know going forward for this course. If you are already familiar with VMware and Linux, you are ready to go with these appendices. If you are new to VMware and/or Linux, these appendices are essential, and we strongly recommend that you read them. Better yet, pop open a laptop and experiment with the items in Appendix A and Appendix B. Similarly, if you know VMware and Linux, but haven't used them in a while, you might want to read these appendices as a refresher. Trust us, you need to know this material for the rest of this class!

If you have seen our Incident Handling Step-by-Step book and are wondering how the book and course relate, note that the book is the outline for the course. Additional material is covered here.

Without further ado, let's dive into the material!

© 2016 Ed Skoudis, John Strand

Incident Handling

- Incident handling is an action plan for dealing with the misuse of computer systems and networks, such as
  - Intrusions
  - Malicious code infection
  - Cyber-theft
  - Denial of service
  - Other security-related events
- Keep written procedures and policy in place so you know what to do when an incident occurs

Incident handling is the action or plan for dealing with intrusions, cyber-theft, denial of service, and other computer security-related events. Your Incident-Handling plan should include hooks to your general Disaster Recovery and Business Continuity plans that deal with fire, floods, and other disastrous events. The scope of incident handling is greater than just intrusions; it covers insider crime and intentional and unintentional events that cause a loss of availability. Furthermore, intellectual property is becoming more important as we move into an information age. Types of intellectual property include brands, proprietary information, trade secrets, patents, copyrights, and trademarks.

The other key point of the definition is the notion of action. Sitting there watching is not incident handling. Identifying an incident is important, but you must act on that information to secure your systems in a timely manner. The best way to act on an incident and minimize your chance of a mistake is by having proper procedures in place. Well-documented procedures ensure that you know what to do when an incident occurs and minimize the chances that you will forget something.

Your incident-handling plans and policies must comply with the applicable laws of your country. We discuss some of these legal aspects in more detail at the end of the day, after we describe the incident-handling process itself.

© 2016 Ed Skoudis, John Strand

Incident Definition

- The term "incident" refers to an adverse event in an information system and/or network...
- ... or the threat of the occurrence of such an event
- Focus is on detecting deviations from the normal state of the network and systems
- Examples of incidents include
  - Unauthorized use of another user's account
  - Unauthorized use of system privileges
  - Execution of malicious code that destroys data
- *Incident implies harm or the attempt to harm*

This slide and the next one are for the purpose of defining what we mean when we use a word like "incident" or "event." Incident, as we use it, refers to actions that result in harm or the significant threat of harm to your computer systems or data. Looking for incidents involves finding deviations from the normal state of the network and systems. There are several important points for an incident handler that flow from this definition. First, because we are dealing with harm or potential harm, our task is to limit the damage. We want to be careful to choose courses of action that do not cause further harm.

Second, your organization may well have a right to redress. We cover this in depth later, but there are criminal and civil law remedies associated with computer incidents. In either case, the incident handler should proceed in a manner that does not preclude use of the evidence gathered in a court setting. A handler does not know in advance whether a given case will go to court. Although only a small fraction of most cases end up in court, you need to treat all of them from the outset as though they may go to court. Don't worry; that's not an enormous burden. It just means doing your job thoroughly and documenting your actions carefully.

© 2016 Ed Skoudis, John Strand

Event Definition

- An "event" is any observable occurrence in a system and/or network
- Examples of events include
  - The system boot sequence
  - A system crash (could be normal behavior for that system)
  - Packet flooding within a network (could be bursty, legit traffic)
- These observable events provide the bulk of your organization's case if the perpetrator of an incident is caught and prosecuted
  - Must be recorded in notebooks and logs
  - Recording the same event in multiple places helps improve evidence—that's corroborating evidence

Events are observable, measurable occurrences in our computer systems. An event is something that happens that someone either directly experiences or that you can show actually occurred. An event is something that you see flash on the screen or that you hear. It can also be something that you know occurred because it was collected in a log or audit file.

At http://www.sans.org/score/incidentforms, you can find forms that help you document the information that should be documented; these forms help alert you to the things you should look for. The forms' copyright allows you to make all the copies you want.

If there is any chance of the incident ending in a court case, having corroborating information is better than a single source claiming that the event happened. For instance, if two people see a message flash on a screen, this fact will likely have more validity in court than if just one person saw it. Further, attackers sometimes use tools to alter or delete their traces in log files. If you can produce two independent sources for the information, your evidence has more validity. This is one reason we push intrusion analysts to become familiar with a large number of log formats. Let's look at an example on the next slide.

© 2016 Ed Skoudis, John Strand

Corroborating Evidence: Microsoft IIS Attack?

Snort output:

```
[**] IIS vti_inf access attempt [**]
06/25-05:36:13.833982 63.209.91.33:4791 -> 10.0.0.13:80
TCP TTL:116 TOS:0x0 ID:6075  DF
***PA* Seq: 0x1CB6779   Ack: 0xB58F0491   Win: 0x217C
```

- Which corresponds directly with:

Log output:

```
[Wed Jun  25 05:36:13 2016] [error] [client 63.209.91.33]
File does not exist: /usr/local/apache/htdocs/_vti_inf.html
[Wed Jun  25 05:36:14 2016] [error] [client 63.209.91.33]
File does not exist:
/usr/local/apache/htdocs/_vti_bin/shtml.exe/_vti_rpc
```

- Incident or event?
  - We must look at environment and context

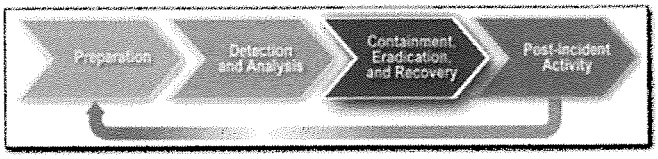
On this slide is a bit of corroborating evidence that can help determine what is happening on our systems and even bolster a case. The main point is that two different systems have captured the same event of interest. The top block comes from a Snort intrusion detection system. It has a rule that causes it to alert if it sees the signature for a particular attack; in this case, it is an attack on the Windows web server IIS. This particular attack is against a weakness in a default script on Windows 2000 IIS servers named vti_inf.

The bottom block comes from a UNIX web server running Apache software. So, there isn't a lot of risk of harm here; a Microsoft IIS attack is unlikely to succeed against a UNIX system running Apache. Some people would classify this as an incident, because the attacker probably did have malicious intent. Others would say that because no harm can be done, it should not be considered an incident. The point, however, is that the intrusion detection system and the web server are completely separate systems and they show the same event. If this went to court, having both logs of the event will make for stronger evidence. Also, when you have multiple sources of information, there is a good chance that you will be able to get data from one that might not be available in another. This is why the intrusion analyst and the incident handler should work hard to learn the types of log files available to them and develop the skills needed to read the logs.

© 2016 Ed Skoudis, John Strand

Executive Summary

- Incident handling is similar to first aid
- The caregiver is under pressure and mistakes can be costly
- A simple, well understood, documented approach is best
- Keep the six stages in mind: Preparation, Identification, Containment, Eradication, Recovery, and Lessons Learned
- Use predesigned forms and ask for help
  - http://www.sans.org/score/incidentforms
  - Forms include Incident Contact List, Identification Checklist, Survey, Containment Checklist, Eradication Checklist, and Comm Log
- Additional materials available are:
  - NIST's *Computer Security Incident Handling Guide*, Revision 2
    - http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf



Law enforcement agents tell story after story of the well-meaning system administrator who ruined the evidence—usually just a couple minutes after the incident. You do need to act, but take time to think.

This story has a crucial point. No one can run so fast that he can outrun a computer with a 3-GHz multi-core processor attached to a Gigabit Ethernet. More importantly, when one is working as root, administrator, or supervisor, many operations do not have an "undo" capability. Several times during this part of the talk, we will draw the analogy between incident handling and first aid. It is a solid analogy; in some ways, first aid is a form of incident handling.

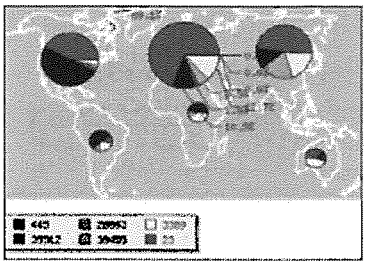
To help you stick to the six-step process, use the forms at the www.sans.org web site. They provide a template for useful information you need to capture during an incident. The free forms at this site include Incident Contact List, Identification Checklist, Survey, Containment Checklist, Eradication Checklist, and a Communications Log.

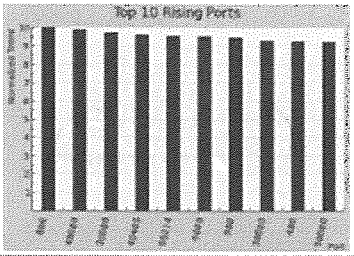
In addition, for more valuable materials, NIST has developed a *Computer Security Incident Handling Guide* that covers the same concepts we do here. It's a solid read and nicely complements this material. You can get it at no charge from the URL listed on the slide. The graphic on the slide is excerpted from the NIST document and summarizes a flow for incident handling fully compatible with the process we cover in this course.

© 2016 Ed Skoudis, John Strand

Share Your Experiences!

- If your corporate policy will allow it, share what you have learned with other incident handlers and incident response teams
  - Attacks against computers are happening everywhere, all the time
  - The bad guys share information; if we incident handlers do not share with each other, they'll stay a step ahead
  - Coordinating your efforts with those on other teams is a critical facet of incident response
  - Do as they told you to do in elementary school: *share*
  - The Internet Storm Center (isc.sans.edu) is a wonderful point of communication, with a handler on duty every day
  - Check out the various "Cons," such as Defcon, Black Hat, and Derbycon





The attacker community cooperates with one another (albeit sometimes in an antisocial manner). They share hacked accounts, exploits, and tricks of the trade.

We often don't share information in the security community. The fact that we may have come under attack seems to be a secret. This will not come as a big surprise, but virtually everyone connected to the Internet comes under attack. Eventually, your organization is bound to take a hit. You can learn from that and you can share what you learn. By doing so, others can learn. If your attackers share and you don't, your organization is outnumbered big time!

So how can you share attack and incident information? You can post something to bugtraq at www.securityfocus.com, or submit information to the handlers' list at the Internet Storm Center (isc.sans.edu). The handlers' list always has an experienced handler on duty, waiting for reports to come in. Each day, the handlers' diary is updated with the latest information about computer attacks. You should check it out! The ISC also displays statistics from the DShield sensor network, which has over 40,000 sensors distributed around the planet gathering information about scans and attacks against various ports. Their world map view shows the countries associated with the source IP addresses of the scans. Additionally, the ISC shows the top 10 rising target ports used in these scans.

There are also a large number of security/hacker gatherings called "Cons" where great (and sometimes scary) information is shared. Following are some of our favorites:

https://www.derbycon.com/

https://www.defcon.org/

https://www.blackhat.com/

http://www.shmoocon.org/

© 2016 Ed Skoudis, John Strand

Course Roadmap

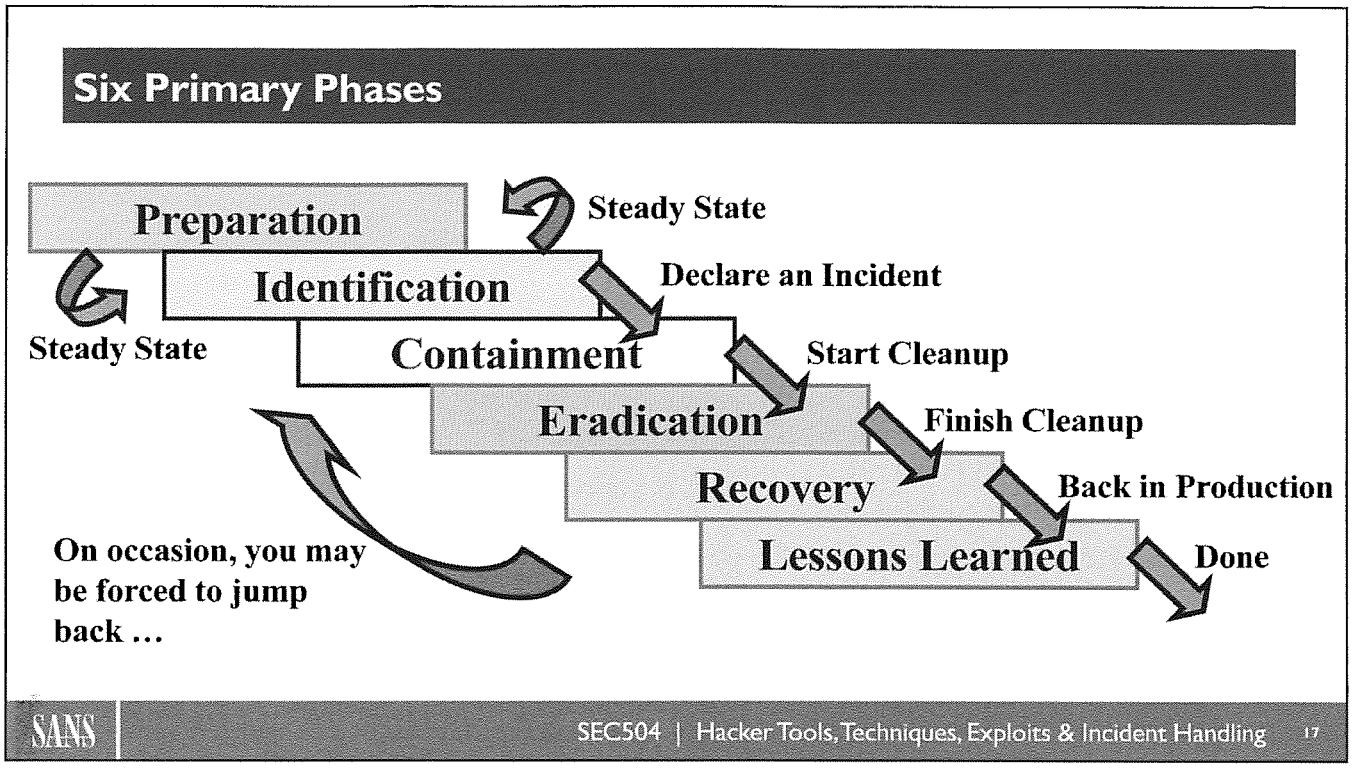
- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. **Incident-Handling Process**
3. Preparation
4. Identification
   - Lab: Windows Cheat Sheet
5. Containment
6. Eradication
7. Recovery
8. Lessons Learned
   - Lab: Enterprise-Wide Identification and Analysis

Here is our outline. We now move to the detailed incident-handling process, the core of this first-day session, and the basis for all of our discussions for the rest of this class.

© 2016 Ed Skoudis, John Strand



The six steps in incident handling are Preparation, Identification, Containment, Eradication, Recovery, and Lessons Learned. The steps serve as a compass or roadmap for the handler; the process is a way for the handler to keep in mind what he is trying to do and the things he needs to do next.

The steady-state, day-to-day practices of most incident handlers are the first two steps: Preparation and Identification. We spend a lot of our time getting ready to fight the next battle and looking for events that could be signs of trouble.

After we identify an incident (that is, events that indicate harm or the attempt to do harm), we move into Containment. Then, the general flow is down the page. You move from Containment to Eradication to Recovery to Lessons Learned. Don't skip steps! Also, we caution you. Try to complete an entire given step in the Containment and later phases before moving to the next phase for a single incident. In other words, for one incident, don't contain it partially on a few systems, and then move to Eradication on those machines while Containment on other systems begins. Do Containment first, and then move to Eradication, and so on. You will likely get organizational pushback on such an approach, but it is the best way to go to successfully handle incidents.

Also, although the general flow of this process is down the page, sometimes you have to jump back up when circumstances change. You might be in the midst of the Recovery phase, when your attacker or malicious code sneaks back in. You've got to be flexible enough to jump back and redo the Containment phase, then Eradication, and then Recovery, for example.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. Incident-Handling Process
3. **Preparation**
4. Identification
   - Lab: Windows Cheat Sheet
5. Containment
6. Eradication
7. Recovery
8. Lessons Learned
   - Lab: Enterprise-Wide Identification and Analysis

Chance truly favors the prepared mind, and this is especially true with incident handling. As mentioned before, the middle of an incident is not a good time to ponder your incident-response process, wonder if there is a command that will enable you to audit an operating system, or figure out how to create a trustworthy forensics image. Therefore, it is imperative to prepare and ensure you have the skills and resources that you need ready to go at a moment's notice.

© 2016 Ed Skoudis, John Strand

The goal of the Preparation phase is to get the team ready to handle incidents

- People
- Policy
- Data
- Software/Hardware
- Communications
- Supplies
- Transportation
- Space
- Power and Environmental Controls
- Documentation

The goal of the Preparation phase is to get the team ready to handle incidents. This slide serves as an overview of the elements needed to prepare the team for an incident; these are actually the fundamentals of contingency planning, and it is advisable to have these basics covered. As we move through the preparation section, we will discuss these items further.

© 2016 Ed Skoudis, John Strand

People

Preparation

- One of the most overlooked aspects of our security posture
- Also, the most easily attacked
  - Via targeted e-mail (spear phishing)
  - Via calls (social engineering)
- Reoccurring training can be a big help
  - Annual training tends to be ineffective
  - Constant reinforcement
  - SANS Securing the Human
- You can also regularly test your users with social-engineering calls and phishing tests
  - Caller ID spoofing is a good test to employ
  - Phishing frameworks, such as sptoolkit and Phishme

In information security, we tend to focus on the easy things—things like IDS, IPS, and AV. These are all technical and can be relatively easily implemented and evaluated. However, in many cases, these technologies are not how attackers target our organizations. Instead, many attackers target what is generally regarded as the easiest attack point: your people.

When attacking an organization, attackers can target users in a number of different ways. The most commonly used ways are via a phone call and through a malicious e-mail.

The best way to prepare for these types of attacks is through constant training and assessment. Many organizations undergo quarterly testing of their technology; the same principle can be applied to people. Once a quarter, either call users via a Social Engineering (SE) campaign and/or utilize a spear-phishing framework to test your user population's susceptibility to clicking malicious links. Projects and services like the sptoolkit and Phishme are excellent ways to create phishing campaigns and track the results.

For more information for training, check out the SANS Securing the Human at:

http://www.securingthehuman.org/

Also, check out the sptoolkit. The code is still active; however, it is currently looking for some people to take over the project:

http://sptoolkit.com/

© 2016 Ed Skoudis, John Strand

Policy – Warning Banners

Preparation

- Establish policy and warning banners ←
  - Warning banners limit the presumption of privacy
- Warning banner must advise the user that:
  - Access to the system is limited to company-authorized activity
  - Any attempt at or unauthorized access, use, or modification is prohibited
  - Unauthorized users may face criminal or civil penalties
  - The use of the system may be monitored and recorded
  - If the monitoring reveals possible evidence of criminal activity, the company can provide the records to law enforcement
- Have legal team review this banner, approving it in writing
- Be careful of local privacy laws, especially in Europe
  - European Data Privacy Directives may impact that crucial line

Crucial!

Don't go flying through this slide because this is such a familiar word.

Warning banners are very important to an incident handler. They make a major difference in the amount of trouble you have to go through to collect and use evidence. Everyone knows you should have them, but if your organization is lax on the implementation, start squawking! This is a battle worth fighting, but be certain to fight in a wise manner. The banner is one tool that can be used to explicitly define your organization's policy on the presumption of privacy. In a world of shades of gray, this is an issue we want to nail down; a handler must know his organization's policy about privacy.

If at all possible, an organization should retain the authority to monitor its networks and systems. That's why that fourth bullet is so important: "The use of the system may be monitored and recorded."

Have your legal team review the language of the warning banner to make sure it will support your monitoring activities and back you up in a court case. Have legal approve it in writing.

This language works well in the United States and several other countries; however, be careful in Europe. Various countries' interpretations of the European Data Privacy Directives may forbid you from monitoring your own data. Your best bet is to get local legal counsel to advise you in such matters.

© 2016 Ed Skoudis, John Strand

Policy – Response Strategies | Preparation

- Establish an organizational approach to incident handling
- Decide generally how you will handle the "big issues" upfront
  - Maintain secrecy or notify law enforcement
    - Most organizations maintain secrecy until they must notify law enforcement
    - That's not always the best policy, though
  - Contain and clear or watch and learn
    - Most organizations have a default preauthorization to contain, but may handle it differently depending on the particulars of the case
- Get management buy-in and signoff of your default practices
  - Document any purposeful deviations from your standard practice when you opt to do so

One thing you want to avoid is having an incident happen and finding yourself in a debate about whether to contain the incident and clean up or to watch the attackers and try to gather more evidence. Likewise, during the time an incident is occurring is a bad time to decide whether your policy is to involve law enforcement or maintain secrecy. The time to make these (career-affecting) decisions is before the incident, keeping senior management and your legal staff apprised.

If you want to consider watch and learn, you should probably spend some time reading about the honeynet project (www.honeynet.org). They probably have the most experience with this of any group on the Internet.

© 2016 Ed Skoudis, John Strand

Notifying Law Enforcement

Preparation

- Reasons you must notify law enforcement
  - Threat to public health or safety
  - Substantial impact to third party
  - Legal requirement based on industry
    - Ex: FDIC, OCC, and Fed Reserve for financial companies
    - Federal Reserve's Suspicious Activity Report at www.federalreserve.gov/boarddocs/press/general/2000/20000619/form.pdf
- You may need to notify the public if PII or PHI is breached
  - Over 45 other states have breach disclosure legislation
  - If you do business in that state (such as having customers there), you may have to disclose a breach
  - Other countries, the U.S. Federal Government, and other states are working on similar legislation
- Optional reasons to notify law enforcement
  - To benefit from criminal discovery process
  - To be a good corporate citizen

This list identifies the reasons you might want to call law enforcement. We believe that too many companies are too timid to call law enforcement. If we want to get tough with attackers, we need to be more willing to get law enforcement actively involved.

The reasons to report an incident include a threat to public health or safety, substantial impact to a third party, or a legal requirement for your industry. You are required to report incidents for these reasons. For example, in the financial services industry in the United States, we have the Suspicious Activity Report (SAR), which the Federal Reserve requires all regulated banks to fill out when various computer attacks occur. Also, note that you may need to notify the public about an incident involving Personally Identifiable Information (PII) or Personal Healthcare Information (PHI), if you do business in a U.S. state that has a breach disclosure law. More than 45 states have such legislation. Other countries, the U.S. Federal Government, and other states are working on similar legislation.

Another reason to notify law enforcement is the selfish one: to benefit from criminal discovery in a court case. The final reason involves just helping the community by making sure the attackers pay for their crimes.

© 2016 Ed Skoudis, John Strand

NOT Notifying Law Enforcement

Preparation

- Reasons not to notify law enforcement
  - Control: There are suddenly two cases
    - Some of their goals are different from your goals (prosecution vs. quickly resuming business)
  - Publicity
  - Risk of continued hacking
  - Risk of equipment seizure and/or interruption to business (while backups are made)
  - If law enforcement agents ask you to do something, you become an agent acting on behalf of law enforcement
    - Legal protections apply to alleged perpetrator regarding unreasonable search

Again, don't let this list frighten you. Although we should be more willing to notify law enforcement, we need to know what the trade-offs are.

The primary down side of reporting to law enforcement involves the fact that two cases are now created. There are control issues and publicity issues with having law enforcement involved. Also, law enforcement may compel you to keep your systems open and exposed to continued hacking, so it can get more evidence. Please understand that this doesn't happen often, but it is a possibility, depending on the nature of the case under analysis. Alternatively, it could seize materials you need to run your business. Or, the investigation could interrupt your business while law enforcement gathers critical data by making backups. Finally, if a law-enforcement officer asks you to do something, you may need a court order to do it, simply because you are acting on behalf of law enforcement. The U.S. Constitution and the laws of some other countries include protections against unreasonable search and seizure. Other countries have similar protections for potential/alleged perpetrators.

© 2016 Ed Skoudis, John Strand

Key Points – Interface with Law Enforcement

Preparation

- Utilize SANS "Interfacing with Law Enforcement" FAQ
  - http://www.sans.org/score/faq/law_enf_faq/
- Develop interfaces enforcement agencies
  - Contact local law enforcement before there is an incident
- Know the types of cases that interest them
  - Ask them... usually, something legally novel or high-value
  - Terrorism cases are of special interest these days
- Join InfraGard – www.infragard.net
- Join High Technology Crime Investigation Association (HTCIA) – www.htcia.org
  - More than 36 chapters around the world
- Join Electronic Crimes Task Force (ECTF) – http://www.secretservice.gov/ectf.shtml
- Report to federal/state/local? Starting at the bottom and working your way up is a good strategy.

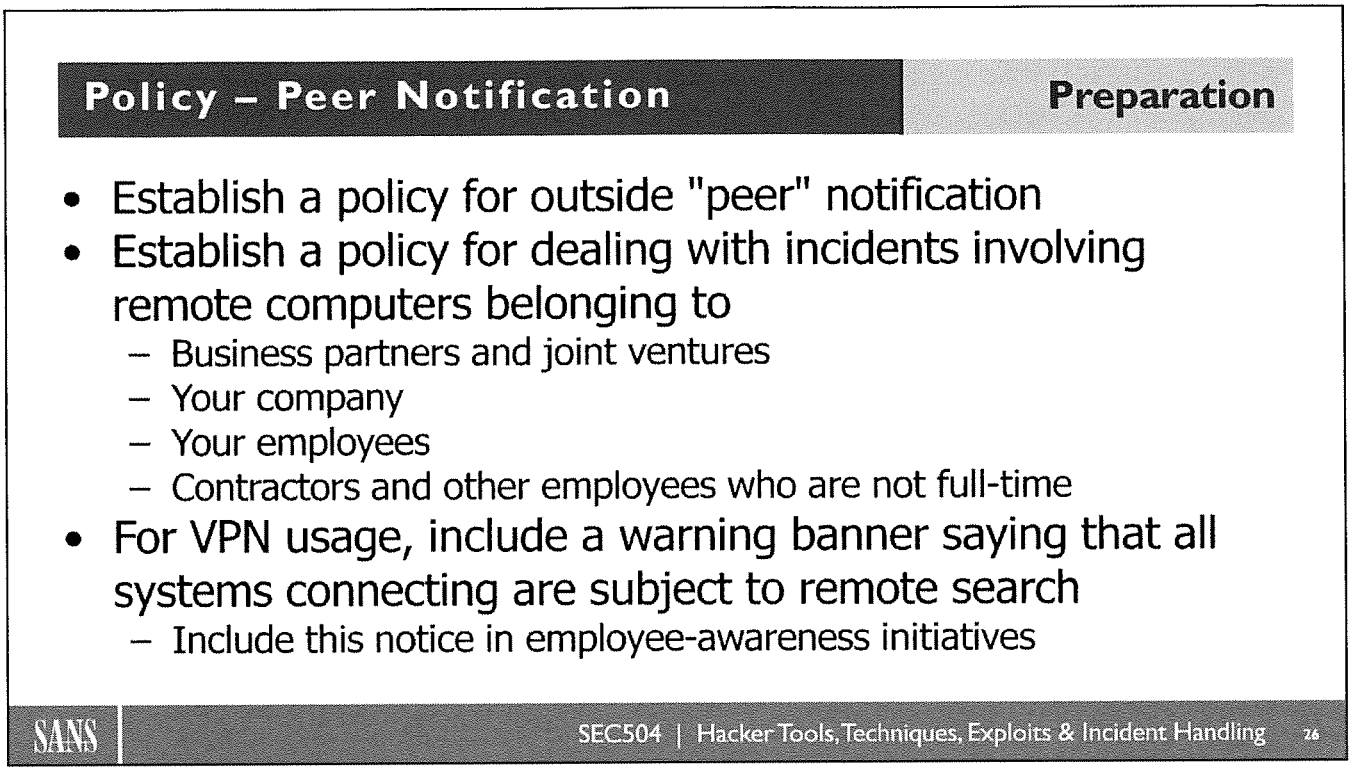
To help sort out the issues associated with interfacing with law enforcement, check out the SANS "Interfacing with Law Enforcement" FAQ at the URL listed on this slide. This FAQ deals with such issues as devising incident response and forensics procedures, maintaining chain of custody, reporting computer crime, and reporting to federal, state, or local authorities.

Folks, this is where you find out what you are really made of and if you are handler material. Interfaces to CIRTs and law enforcement are a long-term investment. It doesn't happen overnight or without effort. These relationships can make all the difference in the world if something bad happens.

Know your FBI agent and your local or state police computer crime officer by name. Get to know him or her through a local chapter of the HTCIA, ECTF, or InfraGard, if such chapters exist in your area. Do a joint exercise and ask questions in advance to try to determine their interests (or what doesn't interest them). Write the answers down so that if they are angry with you later, you can show them that you tried to follow their rules of engagement. Try to exchange PGP keys in advance. You may also want to consider agreeing on a "conventional encryption" tool in case PGP keys become corrupt. In such a case, you can give them the passphrase over the phone; this is still a lot more secure than sending e-mail in the clear.

We frequently get asked whether a handler should report attacks to federal, state, or local authorities. Starting at the bottom and working your way up is a good strategy. Also, leverage the relationships you've made through Infragard and/or HTCIA.

© 2016 Ed Skoudis, John Strand



The unwritten policy on incident notification in some organizations is never to tell anyone anything for any reason. But if there is any chance of the incident spreading and people finding out you had an incident because they were affected, this policy is not ideal.

What happens if the computer is not one that you own, but it has your data on it? The classic example is the employee who takes work home and has a system compromise at home, which involves a business system or even business data stored on his own home computer. Is the employee required to notify your organization? Are you going to do a full backup of that computer? It has three years of Turbo Tax data on it! Are you going to erase that hard drive?

For VPN usage by your employees, include a warning banner invoked upon VPN access that says that all systems connecting through the VPN are subject to remote search by the organization. Include a notice about this search possibility in employee-awareness initiatives. Even though you won't often take this course of action, it is still useful to have it as an option.

What about a consultant who visits your organization and the consultant's laptop is detected scanning the organization's file server? We know what you want to do, but what policy supports that?

What is your organization's road warrior policy? We have met only one person who took encrypting laptop data seriously and who did it consistently, and that was Simson Garfinkel, author of a book on PGP. What if a corporate laptop is stolen? Are you prepared in advance for such circumstances?

© 2016 Ed Skoudis, John Strand

Remain Calm and Take Notes

Preparation

- Remain calm
  - Even a fairly mild incident tends to cause stress
  - Communication and coordination become difficult
- Do not hurry; mistakes can be very costly
- Notes, logs, and other evidence are crucial
- Hand-written notes can be a big help
  - Judges and juries resonate with them
  - The attacker cannot steal them from your machine or destroy them in a Denial of Service attack
  - They help you organize your thoughts and act as a governor on your speed
- If you are going too fast to take notes, you are going too fast!

Whenever people are under stress, communication tends to degrade; this is true even with experienced veterans. Have you ever wondered why folks in professions where communication is life threatening and critical, such as police, fire fighting, rescue, warfighters, and commercial pilots, all use a formalized language? You know what I mean: "Alpha Yankee Zulu, this is Popeye niner, I have a bogey on your six." They adopt a language that has explicit meaning, so errors of interpretation are less likely. They practice speaking that language so they can do so when under stress.

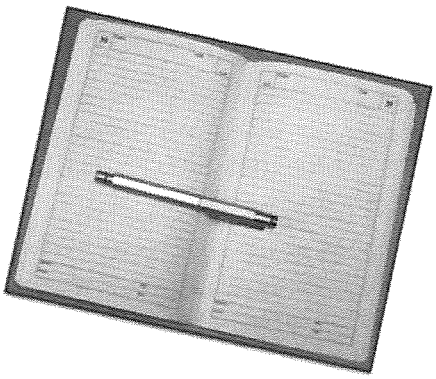
A sure sign that you are in too much of a hurry is when you don't have time to take notes! This is one of the most common, least excusable errors incident handlers make. It is a sinking feeling when you get a phone call and the person on the other end is telling you that the perpetrator of an incident from six months ago has been arraigned. The court date is two months from now, and you realize you can't even come up with the date of the incident. How long should you keep your records? The best answer is to consult your organization's attorney, but in general, keep them as long as possible. We have been contacted by the FBI asking for logs of events that occurred years earlier.

Good record keeping is one of the most important skills that a handler must develop. This is also one of the more difficult things to test, and that is why your practical assignment is a detailed write-up of an incident or related vulnerability. Hand-written notes serve many purposes, including appealing to judges and juries, not being subject to electronic theft by the attacker, helping to organize the handler's thoughts, and governing your speed while handling an incident. In short, if you are going too fast to take good notes, you are just going too fast!

© 2016 Ed Skoudis, John Strand

Maintain Excellent Notes

Preparation



- Take excellent notes in a bound notebook with numbered pages
  - Never, ever, ever rip out a page!
- Your notes may become evidence in court ... 2 years later
- Answer the Who, What, When, Where, Why, and How
  - Who and Why are often the most difficult in intrusions
- Record *all* of your actions!
  - Questions asked, commands typed, systems downed
- Date and timestamp each entry in your journal
  - Include date, time, and name of handler
- A small audio recorder and a *still* camera can be valuable
  - You may want to avoid video cameras, as they could be problematic

The handler must be certain not just to take notes, but to take good notes. The journalist's standard questions of who, what, where, when, and why are a bare minimum. Of the Ws, the hardest ones are who and why in many computer incidents. It is especially important to record all of your actions, including the questions you asked, the answers you received, the commands you typed, the systems you downed, and so on. Make sure you date and timestamp each entry in your journal, including the date, time, and handler's name for each element you write. The forms that are provided in the *Incident Handling Step-by-Step* book have most of the crucial words built in: date, time, location, operating system, and so forth. This can save a lot of time. The forms also remind note takers of what information they should be collecting. Another advantage to forms is you are a bit less likely to doodle!

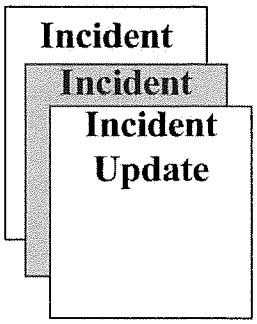
A video might contain far more information about your operation than you want to give away. A single-shot camera can certainly record the scene as you first saw it and is less dynamic than a video. I usually avoid using video cameras in my incident-handling activities, favoring a still camera instead.

© 2016 Ed Skoudis, John Strand

Key Points – Management Support

Preparation

- Develop management support for an incident-handling capability
  - Monthly or quarterly reports on brightly colored paper
  - Graphically illustrate an incident you faced
    - Show jump-off points used in your network
  - Collect historical support
  - If it is a quiet month, collect news articles on computer incidents and other related events, especially in organizations similar to yours...
    - Watch the Handler's Diary at isc.sans.edu and SANS NewsBites
    - Look for similar organizations to yours that are being compromised at http://datalossdb.org/



"I know hackers can be a nuisance, but they can't actually hurt anything, can they? I mean they can't harm anything serious." A senior security manager said this recently. Now we could call him dumb or short-sighted, but unless we reach him in ways he understands, he is going to be very tightfisted with resources for projects like incident handling. What will convince him? Seeing evidence of damage—especially significant harm that could affect his organization's capability to compete—done to organizations just like his is very powerful.

To deal with this issue, write a monthly or quarterly report, on a single page of brightly colored paper. Call it your "Incident Report" and diligently prepare it for management to illustrate what your team has done in the previous month or quarter. If it's been a quiet quarter, list other security incidents in the press and describe how your team is taking proactive actions to deal with similar problems in your own environment.

Graphically illustrating an incident, in essence creating a cartoon, is also a very powerful technique. If a senior executive is able to "get it" and explain how an attack works to her peers, she is more likely to support your effort.

If we do not invest in communication, the only time an incident-handling capability is appreciated is when there is an incident (that scares management) and it is handled well. Getting and keeping management and system administration support are a little like swimming upstream.

Another great online resource for news stories to share with management is the Dataloss Data Base (http://datalossdb.org/). This outstanding site aggregates breach news stories into one easy-to-access site.

© 2016 Ed Skoudis, John Strand

Key Points – Building a Team | Preparation

- Identify qualified people to join the team
- Choose local, centralized, or combination teams
- A multi-disciplinary team is best
  - Security (computer *and* physical)
    - Incident handler(s), forensics analyst, malware analyst
  - Operations (system administration)
  - Network management
  - Legal counsel
  - Human Resources
  - Public affairs/Public relations
  - Disaster Recovery/Business Continuity Planning
  - Union representation (if you are a union shop)
- Obviously, you won't get a full headcount for most of these...
- ...but at least make sure there is someone assigned to you, with a fraction (~10% or more) devoted to the incident-handling cause

One of the challenging problems in building a world-class team is team members who are not hand-selected. Once you get some momentum, everybody wants to get in on the act. When someone wants to forcibly join, or a manager wants to force you to pick someone up on your team, this potentially can be a problem. There is more to a good handler than even desire and technical skill, although those skills are needed.

We recommend an inclusive approach: Anyone willing to study on his own time and able to qualify should have a fair shake at making the team. One solution that seems to work well is a core team and then a larger team that includes your legal and public affairs subject matter experts and security officers or all system administrators. If your organization has one or more unions, then be certain to analyze the contracts with the union before an incident occurs, and you will probably want union representation on the response team.

The demographics of your organization helps determine whether a centralized team is at all reasonable. As a general rule, if you have to get on a plane to handle an incident, the structure you have chosen is not going to work well in practice.

Make sure that your team includes the following disciplines:

- Security (both computer and physical security!). You need skill sets of team members to include incident handling, forensics analysis, and malware analysis. Those skills may be embodied in a single person or divided among multiple people.
- Operations (system administration)
- Network management
- Legal counsel
- Human Resources
- Public affairs/Public relations
- Disaster Recovery/Business Continuity Planning
- Union representation (if you are a union shop)

© 2016 Ed Skoudis, John Strand

Key Points – Checklists and Team Issues

Preparation

- Prepare system build checklists
  - Have most experienced system admins prepare a 5- to 20-page procedure for backing up and rebuilding systems under their control
  - One brief build document per system type
    - For example, standard Windows desktop, standard SAMBA file server, standard IIS web server, standard Apache server, etc.
  - They may have these already ... if so, get a copy and even an image
  - If not, help make it happen
- Establish visibility and a compensation plan for the team
  - Work times and loads vary widely
  - Comp time is important; make sure management understands the need

Our computing environments are complex; no one knows every variant of UNIX and so forth. Although we are trying to make sure you have a solid grounding in the basics of handling systems, memory fades over time. It's useful to have the operations team in an organization prepare brief system build checklists that describe the standard build of each type of system in the environment, in five or twenty pages per system type. Not only will system administrators refer to these documents in their day-to-day work, but incident handlers will also find these documents to be immensely useful in understanding the environment better. If these documents already exist, get a copy for the incident-handling team. If they do not exist, have the incident-handling team work with system admins to help create them during preparation time.

In addition, you may want to get a virtual machine image of your standard builds in the environment, so you can analyze them or at least compare discovered evidence and configuration information against them during an incident. These virtual images can help an experienced handler during analysis.

A large organization with over 10,000 computers is going to rack up some incidents. This can cause the incident handlers to burn out. It is kind of interesting; they tend to burn out just as they become good at their jobs. After training and seasoning, they do a great job on a couple of hot problems and the next thing you know, they are suffering from various stress effects. The solution seems to be a set of things, including rewards, compensation, and time off. This might run afoul of your organizational culture, but consider this: When do incidents occur? They often occur on Friday afternoons at 3:30 PM or later. Do the handlers and administrators go home and wait until Monday to start on the cleanup? No, in almost every case, they stay until the job is done. So, we need to reward these people and let them get some rest.

© 2016 Ed Skoudis, John Strand

Key Points – Team Organization | Preparation

- Define incident-handling team organization
  - On-site/on-location techie handlers
    - Often directly report to a business unit ... with a dotted line to incident handling or even the security team
  - Command post with communications and management organization support
  - Establish a response time baseline
    - Some firm time, between 15 and 90 minutes, depending on the sensitivity of your computing infrastructure
    - Be able to have a technically savvy person onsite within N minutes at all major facilities
    - May not report to incident-handling team, but instead may be part of the business unit

This is one of the most important ideas that came out of the incident-handling research process. In a fairly large incident, technically oriented "action" team members go to the site of the incident, collect data, evaluate the situation, and make decisions (or make recommendations, depending on their site's philosophy). But, this isn't half of the work required to handle a serious incident. A great deal of communications work needs to be done through coordination with other groups, and there are times when the decisions that need to be made must involve upper management.

The observation was made that police departments set up command posts when they need to handle large incidents, and this architecture works well for them. Disaster recovery specialists do the same thing while combating major fires, floods, or tornadoes. The command post needs to be identified in advance, and it should have plenty of communication methods, such as phones, faxes, networks, cell phones, batteries for cell phones, and staff who can collect the information coming in from the field and coordinate that information appropriately. This is the command post team.

In addition to the command post team, you'll also have local techie handlers. To help structure this group, establish a firm timeframe (some firm time between 15 and 90 minutes) for your response team to have feet on the ground at the incident. Make sure you are able to have a technically savvy person onsite within that timeframe at all major facilities. These people may not report to the incident-handling team, but instead may be part of the business unit. Still, make sure that it's part of their job description to support your team.

© 2016 Ed Skoudis, John Strand

Key Points – Emergency Comm Plan | Preparation

- Develop an Emergency Communications plan
  - Create a call list and establish methods of informing people quickly
  - Get a conference bridge number that can be set up with instant notice
  - Print (and laminate if you can) a credit-card sized list of incident-response team contact information
    - Include the name and contact information for each member, and include the conference bridge number
    - Pass them out to everyone on the team
  - Test your call list and tree to make sure it works
    - Try "normal" times and "unusual" times
    - Use these tests to go through an incident scenario

When discussing the Emergency Action plan, we discussed how communications degrade when you are under stress; this is true on the organizational and individual level. So knowing in advance that, in a serious incident, you may need fallback communications, it is possible to plan for this. Telephone communications include voice, fax, and voicemail. Organizations should have call lists to reach key people and "soccer mom style" call trees in case a large number of people need to be reached.

A shared voice mailbox for the core team allows each member access and is strongly recommended. The first member on the scene leaves a message for the others. As additional information becomes available, it is added. This way, the handlers on the scene do not have to stop and continue to give briefings. This works well, but requires discipline to leave messages and discipline not to call the handler on the scene to request an "update."

Finally, consider getting a conference bridge number that can be set up with instant notice. Let your team know the conference bridge access number, but perhaps not the control number necessary to set up a conference. You also should print (and laminate if you can) a credit-card sized list of incident-response team contact information. Include each team member's name, role, phone number, fax number, and PGP key fingerprint (if you are using PGP). Also, make sure to include the incident response conference bridge number. Pass out this credit-card sized list to each member of the team; in fact, give each member several copies for safekeeping.

© 2016 Ed Skoudis, John Strand

Key Points – Getting Access to Systems and Data

Preparation

- The incident-handling team needs to be able to access systems
  - Sometimes without the knowledge of the system admin
- Can be controversial
  - Still, the incident-handling team needs controlled access to computing resources
- Passwords for critical systems and crypto keys
- Strike a bargain with the operations team
  - You'll notify ops management before logging in
  - You'll use only handlers with the skills needed to admin that type of operating system

When I have been the system administrator of a production system, I have never been really comfortable making privileged passwords available to others. However, in an emergency, a handler might need access to critical systems. One organization has a policy that says passwords are kept in sealed envelopes in locked containers. After several years of implementation, the organization has reported that, although sometimes cumbersome, this system has worked well for the company. Note that there is a two-fold responsibility here; the system administrators must make sure the envelopes are kept up-to-date. The handlers must make sure they tread lightly on the systems, keep the administrators up-to-date on any changes they make, and above all, never use a privileged password unless they are qualified on that operating system. One thing that is nearly certain to make an incident worse is having someone who has no clue what he is doing fumbling around as administrator or root.

To help encourage your operations team to give you admin-level access to machines, promise the following and then live up to the promise:

- You will notify the operations personnel on your incident-handling team before you log in with admin credentials.
- You will use only handlers who have enough experience to administer machines of that given type.

Not many of us can change the way our entire organization does business, but we can certainly be responsible for the way that we do business. Encourage people to write down critical passwords and encryption keys and store them safely so they can be accessed if required. As encryption becomes ever more prevalent, an organization must set policy about who owns the secret keys and passphrases and under what circumstances they can be used and accessed.

© 2016 Ed Skoudis, John Strand

Key Points – Point of Contact and Resources

Preparation

- Establish a primary point of contact and an incident command communications center
- In critical sites, establish secured communications
- Set up resource acquisition plans for the teams
  - In advance, you need to get permission
  - During an incident, you may need to quickly procure something; get ready
  - Set aside or get permission to spend $5,000 to $10,000 without going through a multi-month procurement process

One of the functions the command post needs to be ready to provide is rapid acquisition of things needed by the teams. This can span the gamut from a drive to store forensics images to a backup computer, to pizza and Coke for a team that has been on the job for 12 hours, and hotel rooms near the site so the team has a place to crash when exhausted. If you are a manager and you are thinking "yeah, yeah," whose credit card do you think is going to be pressed into service if you are not prepared?

Large organizations in particular can get very rigid about how things can and cannot be procured. It is wise to set up the exceptions to the rules and execute an incident-handling drill where these exceptions can be tested in practice before they are needed.

Remember the test we apply to any of the recommendations: Would I be sorry if I didn't do it? Secured communications can be commercially available, including encrypted pagers and cell phones. They are costly and the phones at least may not give you the quality you really want. That said, if you are in a large organization and the incident involves many millions of dollars, this can be a real comfort. Large or small, there simply is no excuse whatsoever for an incident team that has not established a method for exchanging encrypted e-mail and files, such as PGP or GnuPG.

© 2016 Ed Skoudis, John Strand

Key Points – Reporting Facilities | Preparation

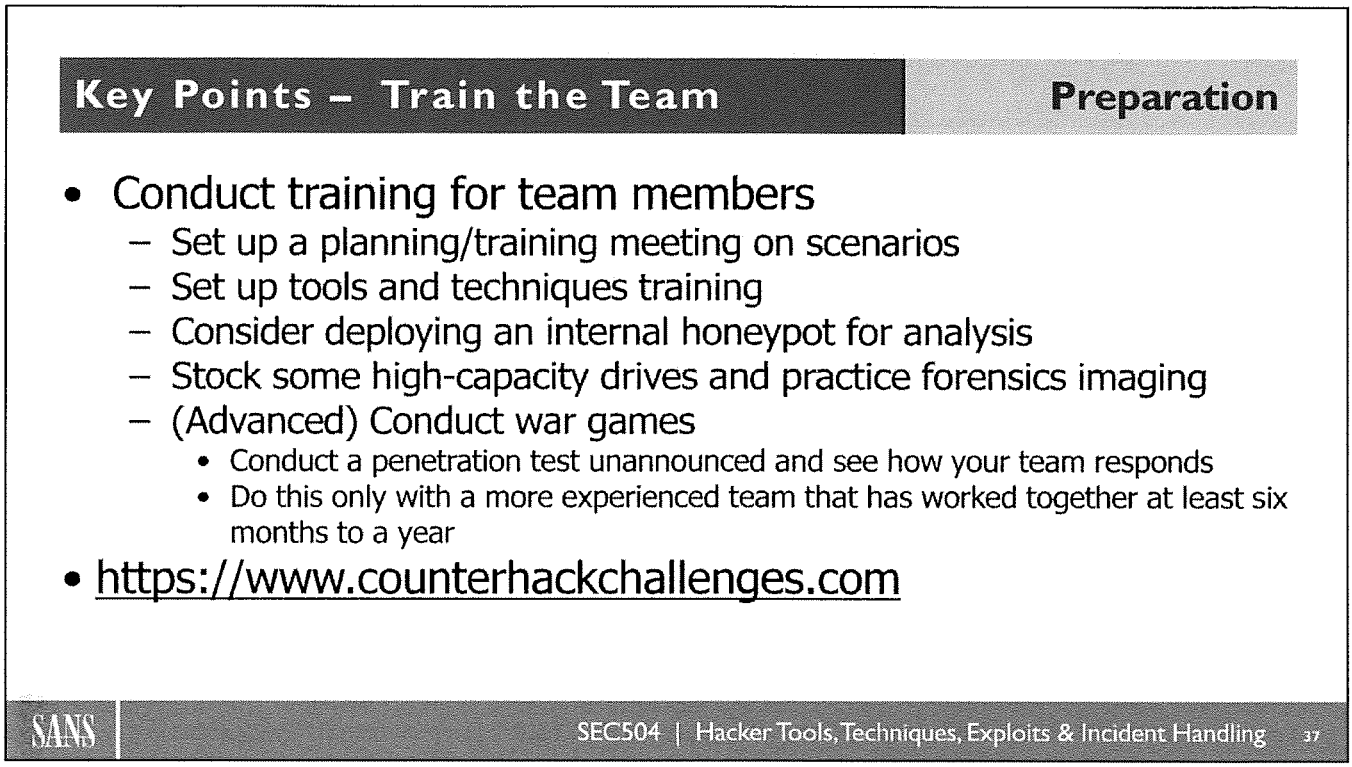
- Provide easy-to-use, convenient reporting facilities for anomalous activities
  - Educate users as they are hired
  - Publish a list of indicators of an incident
  - Use multiple mechanisms
    - Phone reporting: An incident response hotline
    - E-mail: A main incident response mailbox
    - Intranet website devoted to incident handling
  - Reward reporting: Controversial
  - Continually update management
- Establish a war room
  - Should be a place where you can safely display information

Employees—especially network operations people, system administrators, and help desk workers—are the eyes and ears of any organization. One organization rewards employees who report a suspicious (potential) incident with a small cash award. Another writes a small article with a picture when an employee detects and reports an incident. These and other methods of feedback encourage the employees to be alert. These type of things pay off time and time again for the organization.

Make it as easy as possible to report; publish a well-known voice and fax number (your incident-response hotline), an e-mail address, and a web address of an internal web site devoted to incident handling. Employees can experience uncertainty about whether to report an incident; we want them to be as comfortable as possible. Some people prefer to talk to people on the telephone, whereas others are more introverted and prefer e-mail. Give employees a choice. In a major attack, you do not know which of your communication mediums will still be available.

The war room should have a lockable door and a lockable file cabinet. You should have a war room, not a war cubicle. Given the close quarters, it's helpful if the room includes a thermostat for keeping the temperature comfortable for the incident handlers and their equipment. Finally, I prefer to have no windows in my war room, to avoid curious stares from the outside world. By windows, we don't mean the operating system! We mean transparent panes of glass on the walls.

© 2016 Ed Skoudis, John Strand



The #1 training issues are:

- Creating forensics images under fire
- Keyboard skills under fire

To deal with these issues, have your team practice, practice, practice. It is easy to teach the incident-handling process at a general level. What takes persistence and concentration is to hone the skills needed by the on-site, at-the-console, incident handler.

Knowing one method of image creation is not enough; you need to be ready to move to an alternate approach if something goes wrong.

Knowing how to read the audit logs and investigate a file system requires knowing the operating system. This is far harder than being able to read Expert Witness reports, and knowledge comes only with training and practice.

We like to walk into the Computer Incident Response Team room at work and start a drill unannounced; it helps your team stay "combat ready." Also, an internal honeypot can be a helpful tool to hone the analysis capabilities of your team.

You also might want to consider conducting a penetration test of your environment unannounced to see how your team detects and responds to it. However, I caution you: Only conduct such "war games" with a more experienced team that has worked together for at least six months. Otherwise, an inexperienced team may trip over itself and result in negative feelings with such an experiment.

There are also outstanding resources, such as Counter Hack Challenges, which can serve to continuously train your team.

© 2016 Ed Skoudis, John Strand

Key Points – Cultivate Relationships | Preparation

- Coordinate closely with help desks
  - Help desk personnel are often the incident handler's initial eyes and ears
- Pay particular attention to relationships with system administrators and network administrators
  - Involve system administrators in your team
    - Trust your experienced admins' sense of things that "just aren't right"
  - Conduct proactive training
  - Recognize "power" log file reading
  - Encourage regular system backups by sys admins

We need to be candid with one another for a second: Many technical people denigrate the help desk function. They are often entry-level positions and perhaps they do not have the system-programming skills that are developed over time. Handlers that wish to be successful best get down off their high horse. There is no substitute for the thousands of eyes that your users have; it is a sensor network beyond compare. When they see something "funny," they tend to report it to the help desk. Also, if a group is going to try social engineering, it is likely to be tried at the help desk. Investing in your help desk, making sure they are trained to be part of the response process is sound practice!

System administrators and network administrators are the wild card in incident handling. If incident handlers find themselves in a culture where the team is at odds with the organization's system admins, the organization has a real problem and it will manifest itself during an incident. The most probable reason for this tension is if the incident-handling team is primarily drawn from the organization's security department instead of equally from security and operations. Remember, if you do not trust each other during good times, you will not work together well when you are under fire.

You simply can't handle a large incident without system and network administrators, but they are likely to make those critical mistakes that happen in the first five minutes of an incident. The best thing to do is get them involved, get them trained, and make them an integral part of your formal response capability.

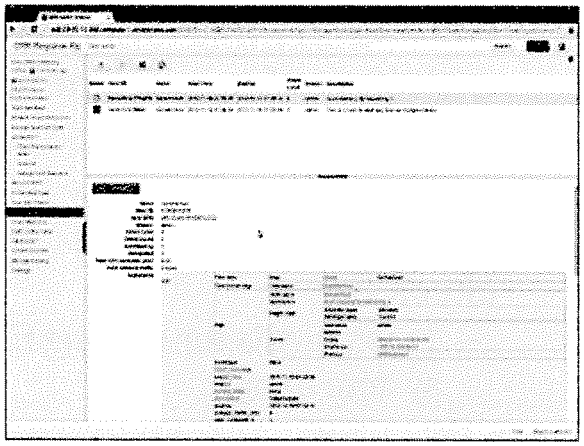
By the way, if a system administrator, especially one that has been around for a while, calls you and says, "Hey, this just doesn't look right," you dismiss that clue at your peril. These folks know their systems and should be rewarded well when they find the subtle clue that indicates a compromise or intrusion that would have otherwise been missed.

© 2016 Ed Skoudis, John Strand

GRR Rapid Response

Preparation

- Maintained by Google
- Free
- Runs on Linux, OS X, and Windows clients
- Remote memory analysis via Rekall
- Python-based agent
- Powerful backend
- Detailed monitoring of clients
- GRR has the ability to pull in-depth forensic artifacts from multiple systems
- Because the pull is asynchronous, it allows you to pull information from computers that are not on the network at the initial request, but rather when they are online again; very good feature for laptops



https://github.com/google/grr

A fantastic tool for performing large-scale incident response and hunt teaming is GRR. Currently, this project is being maintained by Google and is free. It also runs on Windows, Linux, and OS X clients. One of our favorite pieces of functionality in this tool is the ability to perform memory analysis on remote hosts when coupled with Rekall.

It also has the ability to pull and store a wide selection of forensic relevant data from a large number of hosts in an asynchronous manner. This means, if a host is not on the network when a pull of information is requested, it can wait until that system is back on line and connects in to gather the data. This is especially powerful for systems like laptops, which may not be online when the pull request is made.

You can get it here:

https://github.com/google/grr

© 2016 Ed Skoudis, John Strand

Jump Bag

Preparation

- Get a duffle bag and keep it stocked with items for incident handling
- Don't steal from your own jump bag
  - Always have it ready to roll
- Use the following as a check list:
  - Fresh media for holding file system images
- CDs, USBs, and an extra high-capacity hard drive

This slide lists the items most incident handlers feel are needed in a ready-to-go bag, or a jump bag. We discuss these items in detail as we work through the next section of the course, but these are some of the basic tools that you probably want in your ready-to-go bag.

If you do not have a jump bag, you might wish to try an exercise to see how long it would take to assemble one on the fly; you might find it can take weeks.

Remember, never steal from your own jump bag. You will regret it! Odds are, whatever you "borrow" from your jump bag "temporarily" will be the item you require on your next job. Of course, you likely forgot to replace it in your jump bag, so you are out of luck. Therefore, always keep your jump bag intact and replenished.

First off, get some fresh back-up media. I recommend having some blank CDs, USBs, and an extra high-capacity hard drive ready to go at a moment's notice.

© 2016 Ed Skoudis, John Strand

Jump Bag – Software

Preparation

- Binary image-creation software
  - dd, Netcat, and ncat for moving data across the network, Safeback, and so on
- Forensic software
  - Sleuth Kit and Autopsy (free at www.sleuthkit.org)
  - EnCase (commercial software from Guidance Software)
  - Forensics Toolkit (commercial software from AccessData)
  - X-Ways Forensics software (commercial)
  - Others

One of the hardest things to do quickly is to burn CD-ROMs with the core binaries for your organization's operating systems. This is a very low-cost step that you can easily do in advance; CD burners are under US$100 and CDs are less than a dollar. However, it takes time to build these response CDs, and it is a good idea to practice with them before the incident to make sure you can set the path so that the binaries execute off the CD and not from the system's hard disk.

Your software library that you carry with you should include binary image-creation software, forensics software, such as the free SANS SIFT Kit (available either as a VMware image or a stand-alone ISO), the Sleuth Kit and Autopsy tools, or the commercial EnCase tool from Guidance Software and Forensics Toolkit from AccessData, and CDs with statically linked binaries of critical operating-system executables.

© 2016 Ed Skoudis, John Strand

Jump Bag – Additional Software

Preparation

- Diagnosis software you can trust
  - Statically linked binary executables from a write-once media (CD or USB)
  - Attackers often change built-in programs to lie to system admins and hide the attackers' tracks
- Rootkits, by definition, alter the trusted components of OS software to give an attacker stealthy control
  - We advise against the use of bootable Windows PE environments
- Most of them change the hard drive and contaminate evidence
  - Instead, use a good bootable Linux environment, such as SIFT

You'll need solid, trustworthy software when you arrive at the scene. Bootable CD-ROM packages are invaluable, because running tools from a CD-ROM during an investigation is far more reliable, because the attacker will not have altered the CD.

Statically linked binaries don't rely on any libraries on the victim machine. Because attackers might have altered these libraries, we want to use tools that have all libraries built-in.

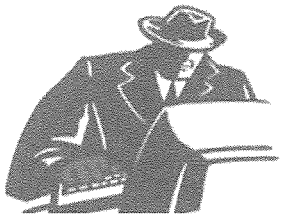
We caution you … avoid bootable Windows CDs for incident handling and forensics purposes. They might be OK for recovery if you have no intentions of doing any sort of evidence collection. They tend to be bad for forensics, because most of them alter the hard drive. If you use them, they will likely alter the hard-drive contents, possibly corrupting crucial evidence.

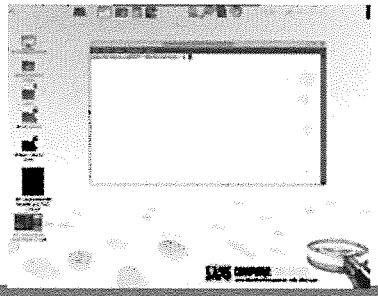
© 2016 Ed Skoudis, John Strand

Jump Bag – Investigative Tools

Preparation

- The SANS Investigative Forensics Toolkit (SIFT) image can be helpful
  - Freely available at http://computer-forensics.sans.org/community/downloads
  - This VMware appliance includes numerous analysis tools:
    - Sleuth Kit
    - log2timeline
    - Wireshark
    - Volatility
    - ssdeep and md5deep
    - Numerous others





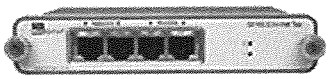
Your jump bag should include an operating-system image that you can use to gather and analyze evidence. One of the best freely available Linux environments for investigations, incident handling, and digital forensics is the SANS Investigative Forensics Toolkit (SIFT), freely available from the SANS Institute. This VMware appliance includes hundreds of different tools you can use to analyze evidence. It includes the Sleuth Kit, a fantastic forensics tool, along with the PTK and Autopsy GUI front ends for Sleuth Kit. It also includes log2timeline, a tool for analyzing the relative times of different events recorded in logs. It includes the Wireshark sniffer for analyzing packets, and the Volatility suite for analyzing memory images. It also includes hashing tools, such as ssdeep and md5deep, along with numerous other tools.

© 2016 Ed Skoudis, John Strand

Jump Bag – Hardware

Preparation

- USB Token RAM device (at least 8 Gig)
- External hard drive (with USB 2 and USB 3 interface and possibly Firewire)
  - 2 Terabyte drives are cheap
- Ethernet TAP
  - Four to eight ports preferable, 10/100/1000 Mbps
  - Don't get a switch... too much work to capture data
  - Taps are preferred, but more costly, as they copy all data, including messed-up frames
    - Several vendors, but NetOptics is popular
- Patch cables
  - At least two straight-through and one cross-over Ethernet
  - USB cable and serial cables for routers and other network equipment



For hardware, make sure to include the following in your jump bag:

- **USB Token RAM device**: These devices can hold data temporarily for moving to a machine for analysis. I frequently use my 8 Gig device to temporarily hold malicious code that I need to analyze.
- **External hard drive**. You can copy data to these, including images of entire drives. Make sure you get a USB 2 and USB 3 interface and possibly a Firewire interface.
- **Small Ethernet TAP**: You don't want a switch, because you cannot easily sniff through a switch. You will likely have to build a small network on the fly and don't want to have to bother with configuring a span port or altering ARP entries so you can gather data through the switch. For incident-handling use, a 4- to 8-port TAP suffices. These devices are designed to be placed into a network and grab all data, without the possibility of a collision that could destroy data in a hub-based Ethernet environment. TAPs can be configured with read-only ports, that can only accept data, preventing the attacker from discovering the incident handler's system through an extraneous packet the machine might send. Inexpensive TAPs can be purchased for less than US$300 or less than $100 in some online auctions.
- **Patch cables**: We recommend bringing two straight-through and one cross-over cable to make sure you can connect in pretty much any environment. You should also bring a USB cable for connecting peripherals. Finally, a serial cable with multiple adaptors (including USB) can be helpful in connecting to and gathering data from routers, switches, and other network equipment (firewalls, Intrusion Prevention Systems, etc.).

© 2016 Ed Skoudis, John Strand

Jump Bag – More Hardware

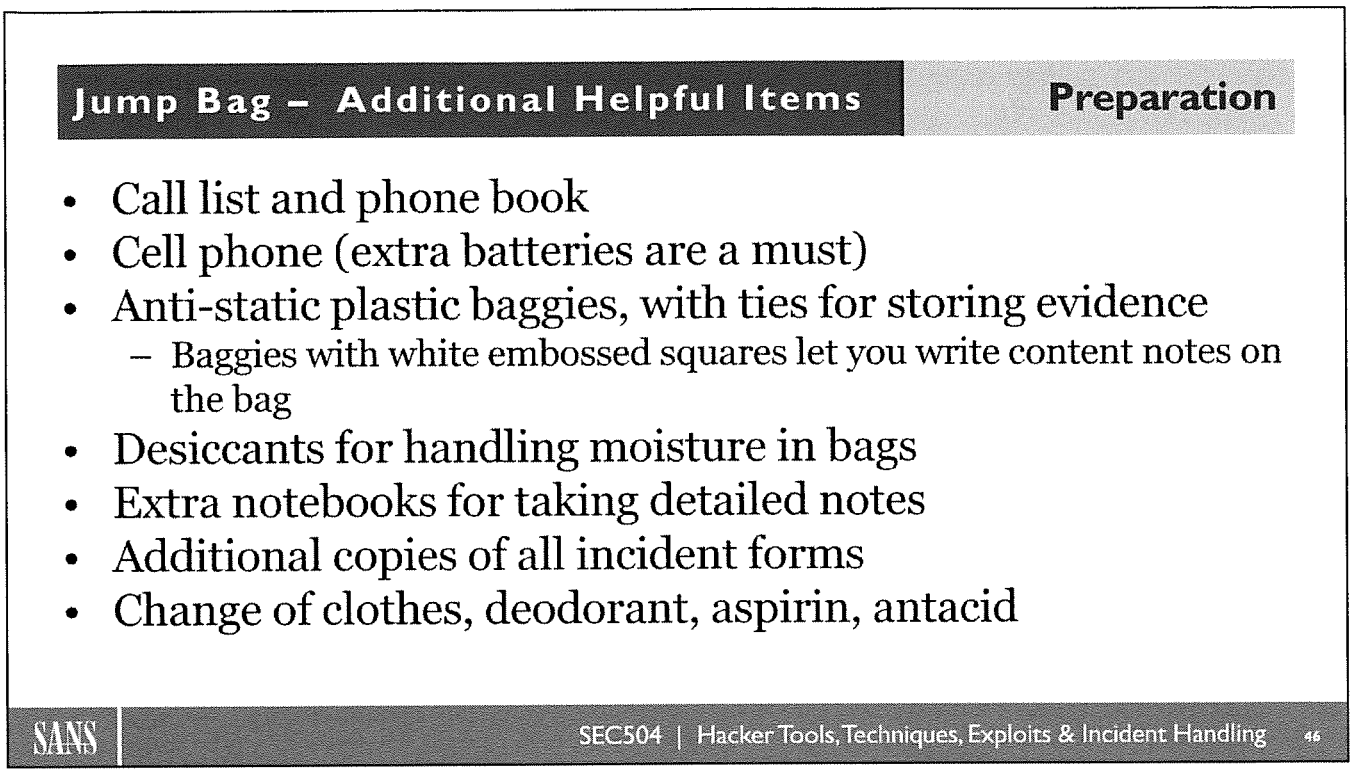
Preparation

- Laptop with multiple operating systems
  - Use the operating systems you have the most experience with and best tuned to your environment, such as Win and Lin
  - Virtual machine software (like VMware, VirtualBox, or Virtual PC) is helpful
  - Large hard drive (at least 1.5 TB, but more if you can afford it)
  - Lots of RAM (at least 16 Gigs)
  - Solid State Drives

Bring a laptop with multiple operating systems. You want to have at least two operating systems that will give you maximum leverage in your organization. Another good option is to use VMware, which is a program that lets you create virtual systems on top of a host OS. With VMware, you can simultaneously run Windows 2000, XP, Vista, Windows 8, Windows 2012 Server, several versions of Linux, and BSD all on the same hardware. You could even throw in Solaris x86 on VMware. This is nice flexibility for an incident handler's analysis. On a single laptop, you can create an entire virtual laboratory.

Solid State Drives (SSDs) are key for quickly analyzing large amounts of data. This is because there is no seek or rotational delay in pulling data from the drive. SSDs can be two to three times faster than traditional spinning drives.

© 2016 Ed Skoudis, John Strand



The following items can be indispensable in an incident, so include them in your jump bag:

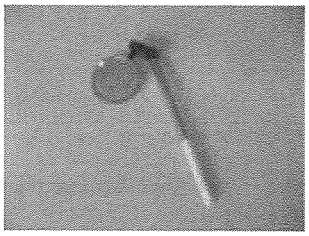
- A call list and phone books are crucial for finding help in your organization when you are under fire.
- A cell phone with extra batteries is useful for communicating for long periods of time.
- Anti-static plastic baggies with ties for storing evidence are helpful in maintaining a chain of custody. I prefer the baggies with a white embossed square on the outside that can be written with a grease pen. That way, I can write evidence/content information right on the bag.
- Desiccants help absorb moisture in your evidence baggies. I carry around a few extra to toss in with a hard drive, just in case.
- You will eventually fill a notebook and use up all of your forms. Bring extras of each just in case.
- Items to support your biological systems, such as a change of clothes, deodorant, aspirin, antacid, and other items. Do not bring liquids in your jump bag; they might leak.

© 2016 Ed Skoudis, John Strand

Jump Bag – Final Items

Preparation

- Small jumpers to alter a hard drive from master to slave
- Flashlight
- Screwdrivers (be careful with airline travel)
  - I like to carry-on my jump bag, and small screwdrivers are likely to be confiscated
- Female-to-female RJ-45 connector
- Extra pens
- Tweezers
- Mechanics' mirrors for looking around corners
- Telescoping "hands" for grabbing small items
- Business cards



These items also help out a great deal. *Small jumpers to alter a hard drive from master to slave can be invaluable.* A flashlight can be useful if you have to crawl behind a rack or look under floorboards.

Screwdrivers might be useful, but watch out for screwdrivers if you plan on flying. They may be confiscated if you try to carry-on your jump bag. I always carry mine on the flight, so it doesn't get lost in baggage. However, it can lead to interesting conversations with the TSA.

A female-to-female RJ-45 connector lets you extend an ethernet cable, while extra pens are useful if you lose your pen.

For work in tight places and with tiny objects, you might find that tweezers, a mechanics' mirror, and telescoping hands for grabbing small items are useful.

Finally, bring extra business cards so you can pass them out to the local people impacted by an incident. It helps them maintain contact with you.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. Incident-Handling Process
3. Preparation
4. **Identification**
   - Lab: Windows Cheat Sheet
5. Containment
6. Eradication
7. Recovery
8. Lessons Learned
   - Lab: Enterprise-Wide Identification and Analysis

This section focuses on the identification of an incident.

How do you detect an incident? The bulk of all detects will come from either sensor platforms or the things people just happen to notice. Sensors include firewalls or intrusion detection systems and system logs, especially with logwatcher software. To increase your chance of detection, consider burglar alarms sprinkled throughout your organization. These include personal firewalls and intrusion detection systems.

People can be your eyes and ears, and they are also spread around the organization. The trick is to give them the training to know that something is wrong and make sure they are aware of the risks and know to whom to report.

© 2016 Ed Skoudis, John Strand

Points to Keep in Mind

- Be willing to alert early!
  - Don't be afraid to declare an incident
  - Even if there is no attack, you still help the organization
- Maintain situational awareness
- Provide indications and warning
- Provide current "intelligence" (up-to-date information) to incident handler
- Fuse or correlate information

There is a pronounced tendency to wait until we are sure something is wrong before we alert. This is death itself; the speed in which incidents occur requires us to field early. So what if it is a false alarm? Use these as training opportunities.

For an incident capability to really perform, there needs to be a constant stream of information. Make sure that you construct your process to support and allow this constant stream.

This is one reason we recommend that you never send only one handler into the field, unless you are very short of handlers. The second team member can maintain communications with the command center. This really helps with the situational awareness component.

© 2016 Ed Skoudis, John Strand

Assigning Handlers

Identification

- Assign a person to be the primary incident handler
  - Select a person to handle identification and assessment
  - Assign him a specific set of events on a specific set of systems to analyze
  - Empower him to escalate if needed
    - Call back to the incident-handling chief if additional resources or expertise is required
- Ideally, assign a helper
  - If you have the resources, it's best to deploy two people to handle each incident to gather evidence

If one person isn't in charge, no person is in charge. For smaller incidents, often of the "would you check this out?" category, there is no need to send core incident handlers. Earlier, we discussed that a recommended practice was to have a core team of well-trained handlers, and have incident-handling skills and training as part of the job for security officers or system administrators. An organization that does this benefits by having multiple levels of trained "fire fighters." However, in such a case, it is important to set up assignments in a way that encourages the system administrator to succeed.

A non-full time handler should be given the assignment in a way that it is clear what is expected of him: The quality of his investigation, what documentation he should produce, and when it is due. It is also important that he knows who he can call if he feels he needs additional support.

Ideally, it's best to deploy two people to handle each incident to gather evidence more thoroughly. Therefore, assign a primary handler, and a helper.

© 2016 Ed Skoudis, John Strand

Control the Flow of Information — Identification

- Enforce a "need to know" policy
- Tell the details of the incident to the minimum number of people possible
- Remind them that they are trusted individuals and that your organization counts on their discretion
- Inform them that they may be required to testify
  - This may scare them... but that's OK

Nothing spreads faster than a rumor! Let's be up front about this. In many organizations, the culture is "we trust our people." That is great. If you go to court, the defense has the right, even duty, to call as many witnesses as required. If 18 of the 20 people called to testify never understood what happened, failed to take notes, and are now recounting the event a year after it occurred, what do you think will happen? A legal disaster might ensue.

Also, a tremendous percent of the time, what you originally think is going on turns out not to be the case. This is common and is an expected part of the incident-handling process; in fact, the process is explicitly designed to handle this. However, if those first clues and theories get published in the newspaper, it can be embarrassing to your organization.

Sometimes, it takes a long time to bring an incident to closure; sometimes, we are dealing with an insider. If someone blabs, that can be the tip-off that ruins your investigation.

Finally, many incidents occur because an individual made a mistake; she did not secure a system as she should, or whatever. This individual may need to be dealt with, but if your team is the one that leaks the info, you will be regarded with mistrust from then on. If that happens, it is difficult to get the information you need to do your job.

© 2016 Ed Skoudis, John Strand

Communication Channels | Identification

- If the computers may have been compromised, avoid using them for incident-handling discussions (e-mail or chat)
- Rely on out-of-band communications
  - Use telephones and faxes
  - Be careful with VoIP, which can be sniffed and played back using a variety of tools (Wireshark, Cain, and VOMIT) if it is not encrypted
- Make sure the team can send encrypted e-mail, such as GnuPG, PGP, S/MIME, and so on
  - Share keys in advance
- Possibly encrypted cloud storage, such as Tresorit or SecureSafe

It is possible to compromise a system, break root, and have a sniffer installed and running in less than 30 seconds. Once in control of a system, the attacker can monitor the e-mail. In such a case, the attacker could reasonably track every move you make if you use the network to discuss it. On that note, you aren't really an incident handler if you don't have the ability to send encrypted and signed e-mail and files, using tools like PGP or Gnu Privacy Guard. People can use your public key to send you sensitive information, and you can use your private key to "sign" instructions so people know they really come from you.

If the computers you'd use to send e-mail or conduct a chat are infected or compromised, pick up the phone and give them a call instead. Be careful with VoIP connections, however, unless you have deployed VoIP with solid encryption. There are a variety of tools that can turn a sniffed packet capture file of a cleartext VoIP conversation into an audio file. This functionality is supported in Wireshark, Cain, and a tool called Voice Over Misconfigured Internet Telephones (VOMIT).

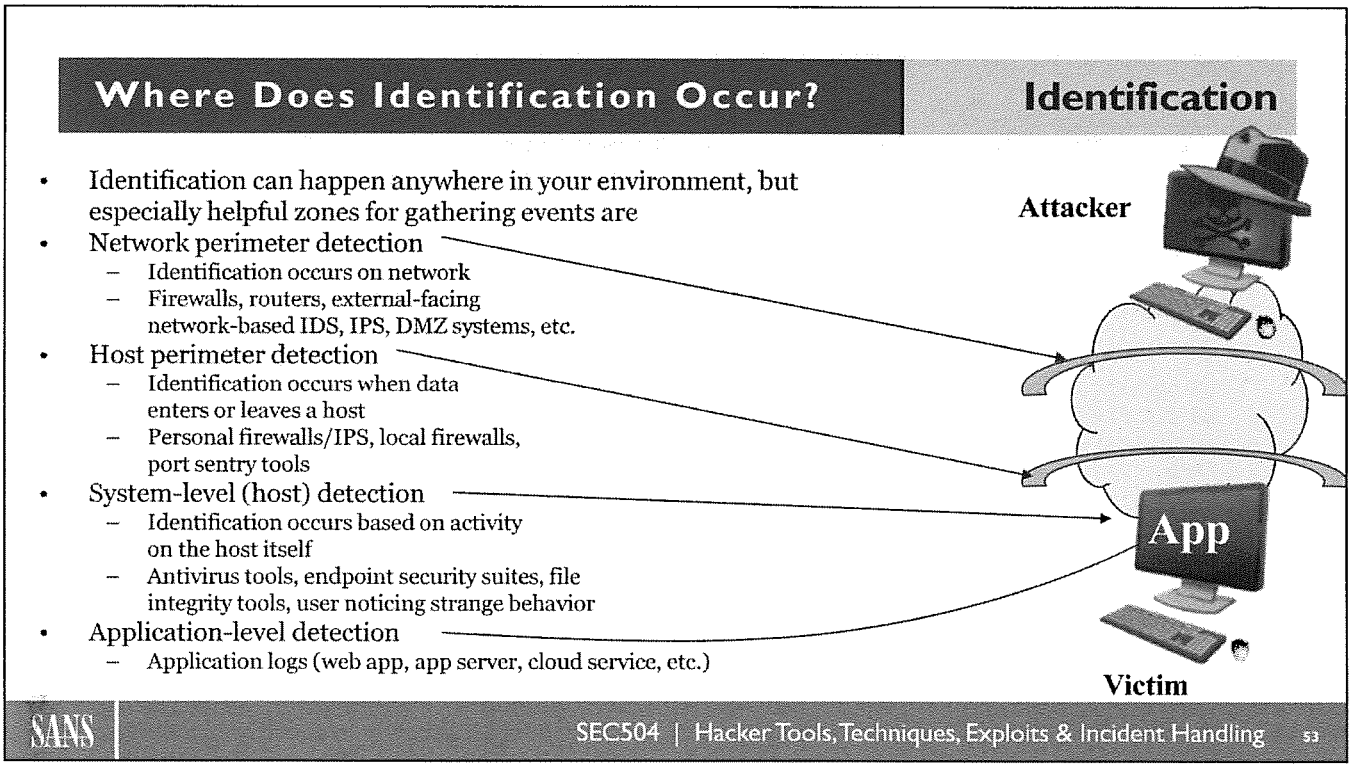
Faxes are a wonderful tool in incident handling. If at all possible, keep a directory with all the fax numbers in your organization and their locations. Make sure your people have actual paper-based fax machines, and not one of those free fax-to-email conversion services. First off, such services send data in e-mail in clear text. Secondly, if your mail servers go down, you won't be able to communicate with your team.

Cell phones are important and handlers should be issued cell phones. In a long incident, several sets of batteries may be needed. On the global scale, the Internet Storm Center has set up an out-of-band network of ham radio operators. With worm class attacks, it is entirely possible that the Internet can be disabled. If that happens and enough people fall over to dial-ups, it could affect the phone system. The reason it could affect the phone system is that there is a finite capacity of circuits. Think about it. Have you ever tried to call someone in an earthquake, hurricane, or any other significant event and received the "I'm sorry, all circuits are busy" error message? Incident handlers must think about out-of-band communications before the event!

Make sure that the incident-handling team has the ability to send and receive encrypted e-mail among the team members. Consider procuring encryption tools, such as the free Gnu Privacy Guard (GnuPG), the commercial PGP, or the various S/MIME solutions available today. Be sure to exchange keys among the incident-handling team in advance!

There are also encrypted cloud storage providers, such as Tresorit or SecureSafe. These providers store your encrypted IR files. Only systems with the proper authentication, encryption keys, and client will be able to access the date. Be careful, this may violate some corporate policies.

© 2016 Ed Skoudis, John Strand



When we consider a high-level view of a network architecture, identification can occur pretty much anywhere in your environment as you gather events. But, to help categorize the various zones in your environment to analyze for evidence of attacks, consider these four levels: network perimeter, host perimeter, host (or system), and application level.

Our network perimeter is monitored by firewalls, routers that generate logs, external-facing intrusion detection systems, Intrusion Prevention Systems, and other machines on the DMZ. These systems can give us earlier warnings about attacks as they monitor our borders with the Internet and other external networks.

The next layer down is the host perimeter, where we monitor activities across each host system's interface, analyzing what the machine is sending out to and receiving from the network. This border can be monitored using personal firewalls and host-based Intrusion Prevention Systems, local firewalls, and port sentry tools.

The next level of detection is host-based, where we monitor the actions on the host systems themselves. Antivirus tools, file integrity checkers, and endpoint security suites often operate at this level. Also, a user noticing strange behavior on her desktop or laptop system falls into this category.

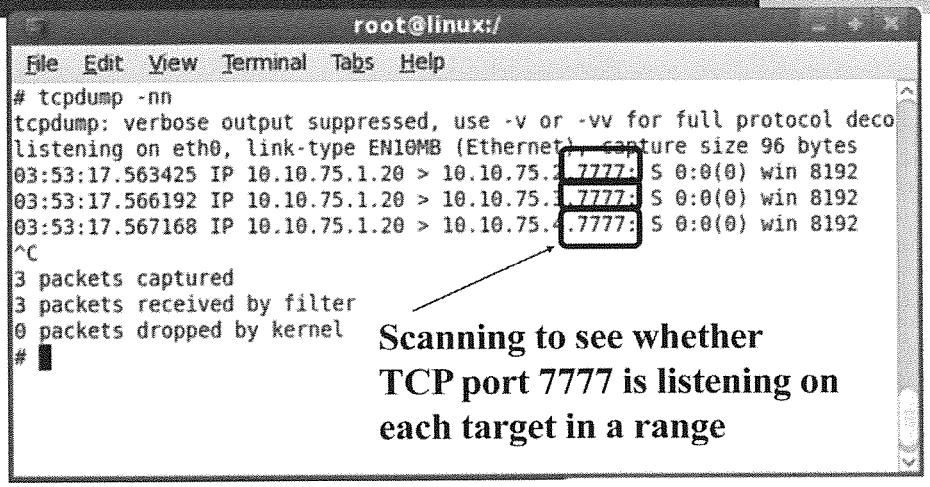
The final level is the application level, which is typically monitored via the logs generated by the application. The application may be a web application, a server-side application used by thick clients, or even a cloud-based service.

Ideally, you want to catch the attack at your perimeter, but sometimes (often, in fact), detection only occurs at the host or application level.

© 2016 Ed Skoudis, John Strand

Network Perimeter Detection Example

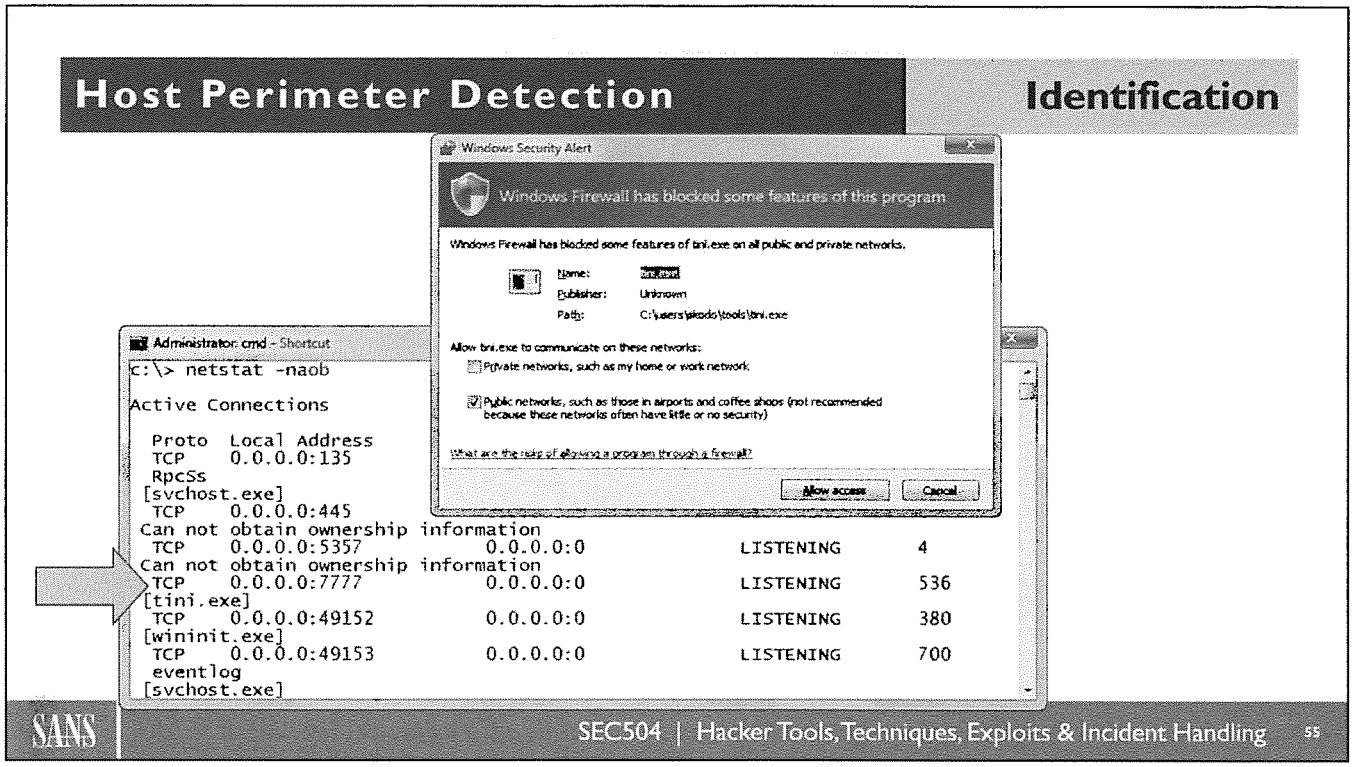
Identification



- TCP port 7777 is used by several malware specimens
- Someone is scanning for the use of that port on 10.10.75.1-3

In the slide's graphic, we see a packet dump (generated by tcpdump invoked not to resolve names with the –n option) of connection attempts to TCP port 7777. Many common Trojan Horse backdoor remote command-shell tools listen on TCP port 7777 by default. We can detect the fact that we have unusual traffic going to this unusual port on a given machine. In fact, note that the attacker tried to connect to destination IP addresses of 10.10.75.2, 10.10.75.3, and 10.10.75.4 on TCP port 7777. This is a network-perimeter detection example, because we saw the packets going across the network.

© 2016 Ed Skoudis, John Strand

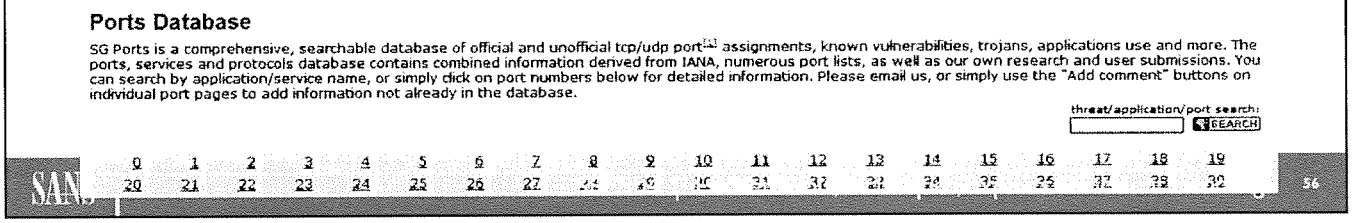


In this graphic, we see some host-perimeter detection examples, with a personal firewall alerting us to the fact that something is trying to listen on the network, namely the tini backdoor. If a user allows it to listen, we still can get detection via the "netstat –naob" command on a Windows machine that indicates that tini.exe is listening on TCP 7777. Note that the –o flag in Windows netstat indicates that we want to see the ProcessID, and that –b flag makes it display the listening EXE and the DLLs associated with it. You see that tini.exe is listening on that port, waiting for a connection. At the host perimeter, we can get more information about what is happening, but our global view is certainly less than with network perimeter detection.

© 2016 Ed Skoudis, John Strand

Analysis of Perimeter and Host Perimeter Detects — Identification

- Look up the service (port list)
  - Internet Assigned Numbers Authority (IANA)
    - http://www.iana.org/assignments/service-names-port-numbers
- Does the destination host run the service? Are you sure? Is it included in your enterprise asset list?
- Could it be a backdoor? A service invoked by an attacker?
  - A useful port list for legit and malicious use of ports is available at http://www.speedguide.net/ports.php



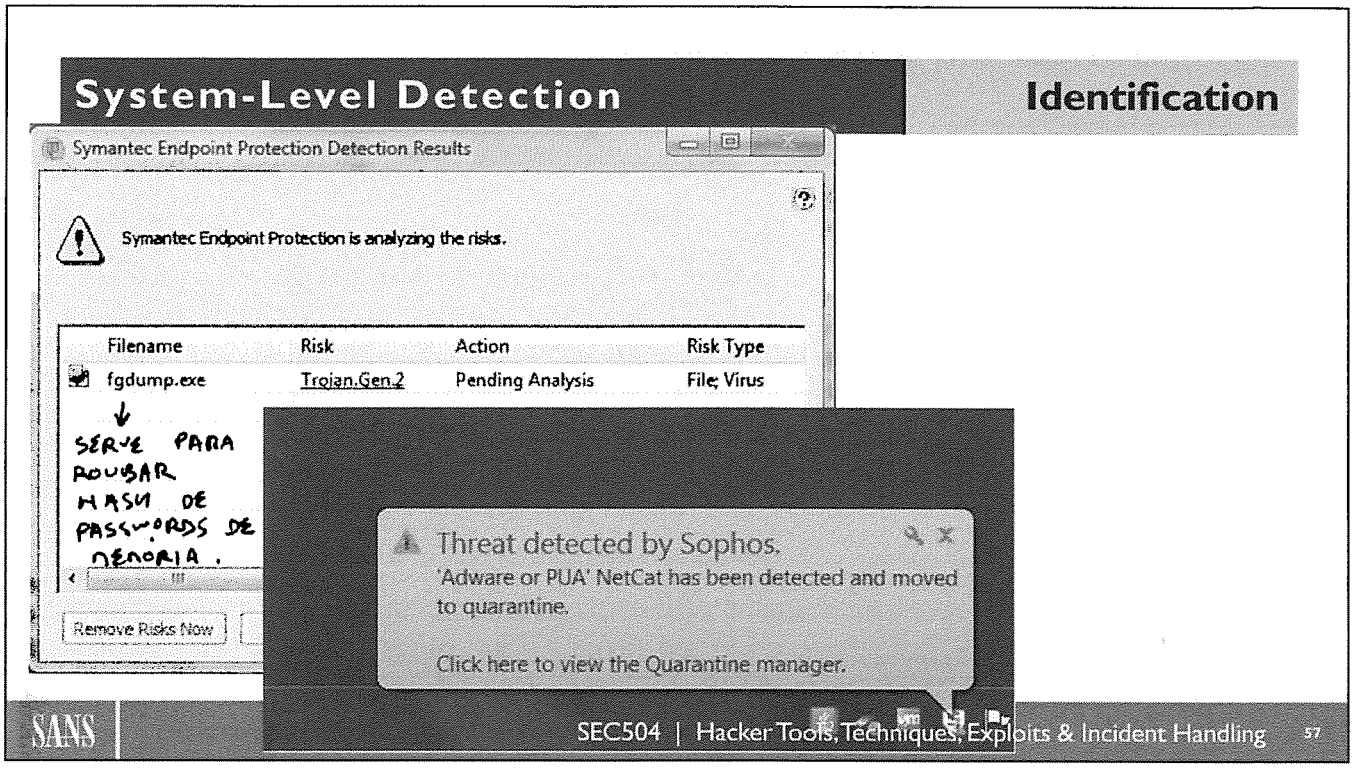
For network perimeter and host perimeter analysis, when you determine a listening port number, you should look up the port to see its official assignment, as well as potential malicious use of that port. There are a couple of relevant port lists. The official port list is maintained by IANA, and you may also want to consult a port list for commonly used Trojans and or malicious code.

Can you be sure that the packet detected at the perimeter is what the port list says? No, of course not; Trojans operate on TCP port 23, the telnet port and many ports have more than one interpretation. However, it is an important start.

Does the intended destination run the service that the packet had in its destination port field? If not, this probably doesn't indicate a good job of reconnaissance on the attacker's part, and the risk is probably very low. On the other hand, if it looks like the attacker knows what she is looking for, you may be in some trouble.

How can you find out if the service is running? Try "`lsof -i`" on UNIX and also "`netstat -a`" on UNIX or Windows. Also, you can scan the system with a port scanner like Nmap, but this may not be reliable. Some attacker software only answers specially formatted queries.

© 2016 Ed Skoudis, John Strand



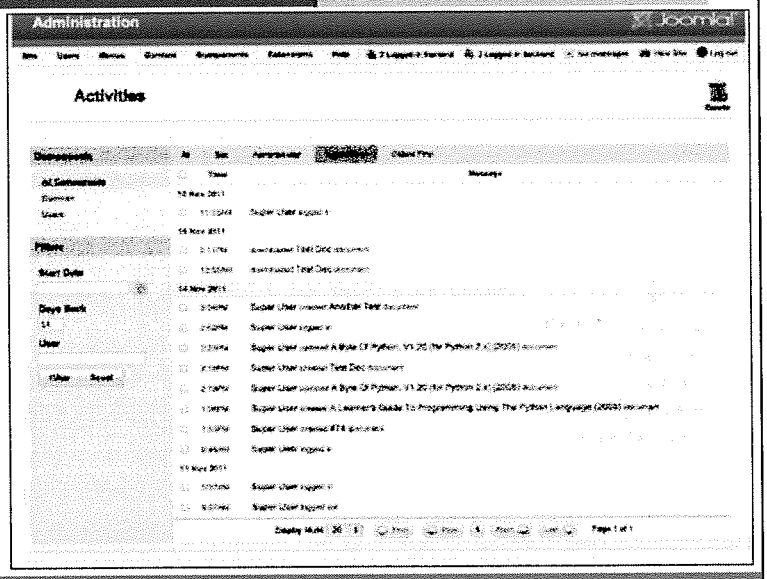
This graphic shows system-level detection via an antivirus tool and an antispyware tool, both of which alerted when fgdump and netcat tried to run. The redundant layers of protection are a good idea, and we get even more information about what's happening at the system level.

© 2016 Ed Skoudis, John Strand

Application-Level Detection

Identification

- Application logs are especially useful from
  - Web apps
  - App servers for thick-client apps
  - Cloud-based services
- Particularly useful data
  - Dates
  - Timestamps
  - Users (especially admins)
  - Actions and transactions, including user input variable values



At the application level, incident handlers can analyze application logs to get a feel for unusual activity that could be associated with an attack that sometimes cannot be discerned by looking at the other layers of identification (network perimeter, host perimeter, and host-level). Such application logs are especially useful when gathered from web applications, application servers that support thick-client applications, and cloud-based services.

Incident handlers should make sure that they have access to such application log data. For the most important applications in their organization, they should also check in advance to ensure that the data includes useful elements of each action taken within the application. Vital data elements, such as dates, timestamps, users (especially administrative level accounts), and actions/transactions, should all be recorded for later analysis. For logged actions and transactions, it is especially helpful if the logs record all user input variable values, a frequent avenue of application-level attack.

See the following resources for reviewing logs for other commonly used applications:

- Oracle: http://docs.oracle.com/cd/E27559_01/admin.1112/e27239/audit.htm
- Apache: http://httpd.apache.org/docs/current/logs.html
- IIS: http://support.microsoft.com/kb/324091

© 2016 Ed Skoudis, John Strand

Identification Across All Levels

Identification

- Ideally, you want to detect attacks at your network perimeter
- Unfortunately, some attacks are stealthy and are detected only after infiltration occurs
- Many incidents are identified only when another site detects **your** site attacking them
- This can cause your site to be blocked or posted on a "shame" web site
- That's why you want identification capabilities at all four levels: network perimeter, host perimeter, host level, and application level

If you visit the web site isc.sans.edu/top10.php, you notice a top-ten attacker's list of IP addresses. From time to time, we get irate e-mails from the owners of such an IP address until they realize the implications of being one of the Internet's ten least wanted.

Many sites create ACLs directly from lists like this, so this could put an aspiring "dot.com" out of business faster than no business plan and no product.

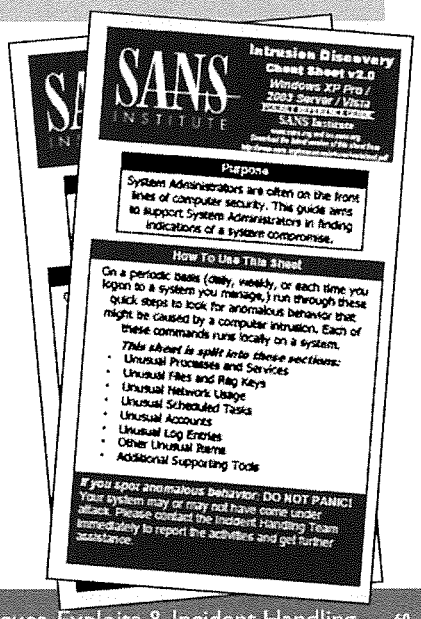
For that reason, incident handlers should make sure that they have access to attack-identification information across all the four levels we discussed: network perimeter, host perimeter, host level, and application level.

© 2016 Ed Skoudis, John Strand

Suspicious Events – Cheat Sheets

Identification

- SANS Intrusion Discovery cheat sheets can be helpful
  - Designed for system admins to spot trouble and call incident-handling team for help
  - 1 page for Windows and 1 for Linux, each a tri-fold
  - Available at http://pen-testing.sans.org/resources/downloads
  - Also included on the Course USB in the Cheat_Sheets directory
  - Free ... make as many copies as you'd like; just don't sell them



To help improve the identification process inside of organizations, SANS created its Intrusion Discovery cheat sheets for Windows and Linux, which are freely available online at the URL on this slide. They are also included on the Course USB, located in the Cheat_Sheets directory. These sheets are designed to educate system administrators in actions they can take to look for anomalous behavior on their machines, including unusual processes, files, network usage, and so on. If system administrators spot trouble, the sheets instruct them to call the incident-handling team (you). Put your name and phone number on a sticker, affix it to the front of the cheat sheets, and pass them out to all of your system administrators.

We'll spend much of the rest of this class (Days 2–5) analyzing various events triggered by each attack we discuss. This initial list is an overview of the details we'll encounter for the rest of this class.

© 2016 Ed Skoudis, John Strand

Obvious Limitations

Identification

- No set of actions can detect every attack, but we shoot for the most common signs
  - Careful attackers using highly stealthy tools will be able to fly under the radar screen
  - But we'll still catch many attackers
  - Even the best attackers sometimes let down their guard
- These sheets expect system admins to know the "normal" state of their systems
  - So they can spot abnormal events
  - Cheat sheet tells them what to get familiar with and where to look for deviations from the norm

Our cheat sheets face some obvious limitations. First, no set of simple-to-perform commands is going to find every single attack. If the bad guy is especially careful to cover his tracks, and employs extremely stealthy tools, we won't be able to detect his presence. Still, many attackers aren't all that careful, and even some of the most powerful tools leave tracks that we can spot. Even the more sophisticated bad guys accidentally leave a few interesting tidbits for an observant system administrator to discover.

Another major limitation associated with these cheat sheets is that they require the system administrators to know the "normal" state of their systems. The cheat sheets identify common areas of deviation from normal that a knowledgeable system admin can spot. However, without a good gut feel of the normal status, these techniques won't work. Therefore, over time, by exercising the tools and techniques covered in the cheat sheets, system admins gradually grow more comfortable with what is normal. After learning the normal state, they can better spot anomalies.

© 2016 Ed Skoudis, John Strand

Windows and Linux Specific Cheat Sheet Elements

- Have the system administrators look for unusual
  - Processes and services
  - Files
  - Network usage
  - Scheduled tasks
  - Accounts
  - Log entries
  - Other unusual items
  - Additional supporting (third-party) tools
- Let's look at Windows cheat sheets in more detail
  - The Linux cheat sheet is discussed in the appendix (titled "Intro to Linux")

Beyond those common elements, the cheat sheets break down the specific technical activities into eight sections. We need administrators to periodically look for unusual processes, files, network usage (including TCP and UDP ports, as well as promiscuous mode where possible), scheduled tasks, and accounts.

We also need them to look at log entries for strange activities. Unfortunately, there are thousands of log entries that could be a sign of attack, yet we are confined to a single page. Therefore, we've listed a handful of the most common log items that might indicate an attack. We also list a handful of other unusual items an administrator should look for.

Finally, we list some supporting, third-party tools that go beyond the base operating system install to help secure the system. These tools are immensely helpful, and some of them should have been built into the operating system from the start. By adding them, we can significantly improve a system administrator's ability to view the status of a machine. NOTE: If you don't want your System Administrators to install these items on their machines, make sure you delete this section from the Cheat Sheet before distributing it to them. We put this item on the back panel of the tri-fold so that it can easily be omitted from your copying process without looking strange.

To get into the details of these cheat sheets, we'll look at the Windows cheat sheet. We look at the Linux cheat sheet during the *Intro To Linux* section in the appendix.

© 2016 Ed Skoudis, John Strand

Windows Cheat Sheet

- The latest cheat sheet applies to Windows XP Pro through Windows 8
  - The cheat sheet is on the Course USB, called winsacheatsheet_2.0.pdf
  - For earlier and less powerful versions of Windows (Windows 2000, XP Home, Vista Home), we have win2ksacheatsheet.pdf
  - There is also an older version on the USB called winsacheatsheet_1.4.pdf
    - That version is just for backward-compatibility purposes

On the Course USB, in the Cheat_Sheet directory, there are several versions of the Windows cheat sheets. The latest version, called winsacheatsheet_2.0.pdf, applies from Windows XP Pro up to and including Windows 8. These versions of Windows include several new tools for analysis and troubleshooting built-in, of which the cheat sheet takes advantage.

For earlier versions of Windows, including Windows 2000 Pro and Server, and less powerful versions of Windows, such as XP Home and Vista Home, we have included another version of the cheat sheet called win2ksacheatsheet.pdf. Given the limitations of these operating systems, this cheat sheet does not include as many commands simply because these operating systems don't have as useful software for analysis built-in.

Finally, some organizations still rely on the earlier version of the cheat sheets: version 1.4. We keep that one on the USB for backward-compatibility purposes.

For this class, we'll look at the latest and greatest version of the Windows cheat sheet (v. 2.0).

© 2016 Ed Skoudis, John Strand

General Approach

Windows Cheat Sheet

- On the Windows cheat sheet, we generally show the admins how to check a given item in the GUI
- Followed by one or more methods for checking the same item at the command line
- It's good to have multiple methods for checking the same thing
- It is often better to start with network connections then work back through the processes, files, and services associated with those connections

Generally speaking, on the Windows cheat sheet, we describe how to check a given item using the GUI first, followed by one or more methods for checking the same thing at the command line.

We offer multiple approaches for checking things for a few reasons. First, some admins are more comfortable with the GUI, while others prefer the command line. Next, the command line lends itself better to scripting, so system admins who want to write scripts to analyze their machines have a starting point for doing so. Also, with multiple methods for checking, the cheat sheets can help function as an educational tool, showing admins some of the features of their systems that they might not have known existed.

When working through the incident response process, it is often better to start with suspicious network connections discovered via the suspect system or netflow data and work back through the associated services and files.

© 2016 Ed Skoudis, John Strand

Unusual Network Usage

Windows Cheat Sheet

- Look at file shares and make sure each has a defined business purpose
  ```
  C:\> net view \\127.0.0.1
  ```
- Look at who has an open session with the machine
  ```
  C:\> net session
  ```
- Look at which sessions this machine has opened with other systems
  ```
  C:\> net use
  ```
- Look at NetBIOS over TCP/IP activity
  ```
  C:\> nbtstat -S
  ```


To look for unusual network activity on a Windows machine, we start out by looking at available shares using the "net view \\127.0.0.1" command. This action will show all file shares on the local machine. System administrators should check to make sure each share has a defined business need.

Additionally, administrators can see if anyone is connected to these shares by running the "net session" command. The output of this command shows any NetBIOS and/or SMB connections associated with file and print sharing and other activities.

The "net use" command shows whether this local machine has made any NetBIOS/SMB connections to other systems. "net session" shows connections *to* this machine, and the "net use" command shows connections this machine has *initiated*.

Finally, we have the administrators run the "nbtstat –S" command to focus on NetBIOS over TCP/IP activity. These connections and shares are likely included in the results of the earlier commands on this slide, but a final nbtstat check couldn't hurt. The –S indicates that we want to see systems connected to our machine, listed by IP address.

© 2016 Ed Skoudis, John Strand

More Unusual Network Usage

Windows Cheat Sheet

- Look for unusual listening TCP and UDP ports
  `C:\> netstat -na`
- For continuously updated and scrolling output of this command every 5 seconds
  `C:\> netstat -na 5`
  - The –o flag shows owning process id
  `C:\> netstat -nao 5`
- The –b option makes netstat show the EXE and associated DLLs using the network
- Also, the built-in Windows firewall settings can be inspected by running
  XP/2003: `C:> netsh firewall show config`
  Vista-Win8: `C:> netsh advfirewall show currentprofile`

Beyond the built-in Microsoft SMB and NetBIOS components, we also need to look at TCP and UDP activity. The "netstat –na" command shows listening and active TCP and UDP ports. By putting a number n after this command, Windows continuously runs the command and updates the display every n seconds.

An additional useful item is the –o option for netstat (as in "netstat –nao"). This flag displays the owner process ID associated with each listening TCP and UDP port.

With the –b option, netstat shows the EXE using the port and the DLLs that it has loaded to interact with the port.

For the netstat command to be useful, of course, the system administrator must have a good feel for the normal TCP and UDP activity of the machine.

Finally, an administrator can dump the detailed configuration of the built-in Windows personal firewall by running the following command:

On XP or Windows 2003: `C:\> netsh firewall show config`

On Vista through Windows 8: `C:\> netsh advfirewall show currentprofile`

© 2016 Ed Skoudis, John Strand

Unusual Processes — Windows Cheat Sheet

- Run Task Manager (Start→Run... and type taskmgr.exe)
  - Look for unusual/unexpected processes
  - Focus on processes with user name "SYSTEM" or "Administrator"(or users in the Administrators group)
- From a command line

```
C:\> tasklist
C:\> wmic process list full
```

- To get parent process ID Information

```
C:\> wmic process get name,parentprocessid, processid
```

- To get command-line options and DLLs

```
C:\> tasklist /m /fi "pid eq [pid]"
C:\> wmic process where processid=[pid] get commandline
```

- **Beware of Base64 encodings**
- **There are great Base64 decoding tools online**
  - **https://www.base64decode.org/**

To look for unusual processes on Windows, the cheat sheet describes running the Task Manager tool by going to Start→Run… and typing "taskmgr.exe."

From the command-line perspective, admins can also view which process are running on the machine by using either of the following commands:

```
C:\> tasklist
```

```
For more details, you can run tasklist verbosely as
C:\> tasklist /v
```

Or

```
C:\> wmic process list full
```

Of course, this requires the system administrator to know what processes are supposed to be running on the machine. Armed with a knowledge of the norm, she can then spot deviations.

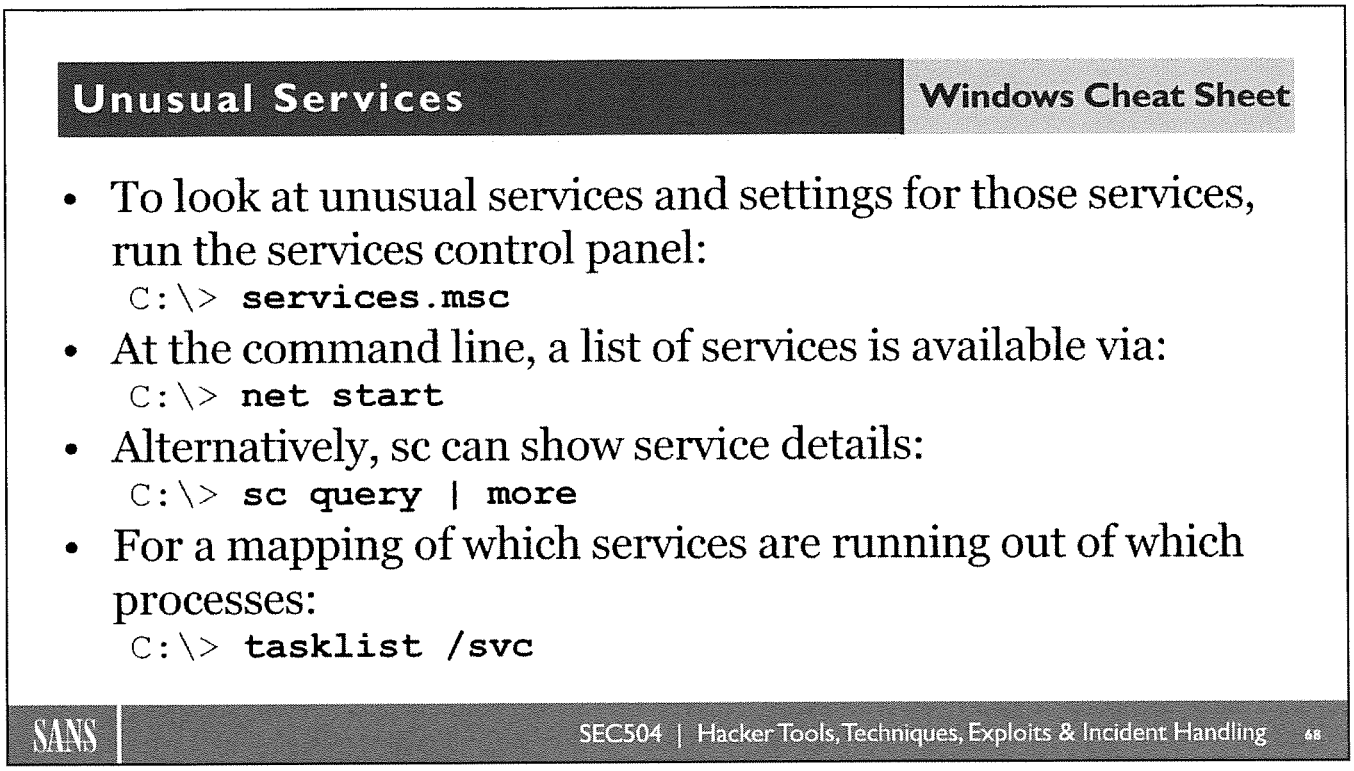
You can also get parent process ID information:

```
C:\> wmic process get name,parentprocessid, processid
```

And, you can get command-line options and in-use DLLs:

```
C:\> tasklist /m /fi "pid eq [pid]"
C:\> wmic process where processid=[pid] get commandline
```

© 2016 Ed Skoudis, John Strand



The services control panel GUI, which shows various services and their status, can be invoked by typing at the Start→Run... box or the command prompt:

```
C:\> services.msc
```

Furthermore, at the command line, to get a list of running services, you could use either of the following commands:

```
C:\> net start
```

The sc command provides more detail of the status of each service:

```
C:\> sc query | more
```

Finally, to see which services are running out of each process on your system, you could run:

```
C:\> tasklist /svc
```


© 2016 Ed Skoudis, John Strand

Unusual Reg Key Entries

Windows Cheat Sheet

- A system administrator can look for strange settings in the registry keys associated with starting programs at system boot or when a user logs on:
  - HKLM\Software\Microsoft\Windows\CurrentVersion\Run
  - HKLM\Software\Microsoft\Windows\CurrentVersion\Runonce
  - HKLM\Software\Microsoft\Windows\CurrentVersion\RunonceEx
- These items should be inspected under both HKLM and HKCU
- They can be analyzed with the regedit GUI
- The Autoruns utility is an outstanding tool for pulling Auto Start Entry Points (ASEs)
- Or, the reg command at the command line can also be used to query the values of these settings

```
C:\> reg query hklm\software\microsoft\windows\currentversion\run
```


A lot of today's malware (such as bots, backdoors) are activated using a registry key designed to execute programs when the system boots up or when a user logs on. A diligent system administrator who suspects system compromise should check out the values assigned under these registry keys.

One of the best tools for reviewing the Auto Start Entry Points (ASEs) on a Windows system is autoruns.exe from Microsoft. It can be found here:

http://technet.microsoft.com/en-us/sysinternals/bb963902.aspx

The following registry keys execute programs as a system boots up or as a user logs on:

HKLM\Software\Microsoft\Windows\CurrentVersion\Run
HKLM\Software\Microsoft\Windows\CurrentVersion\Runonce
HKLM\Software\Microsoft\Windows\CurrentVersion\RunonceEx

These registry settings should be checked under both HKLM (as shown above) and in the HKCU Hive.

These items can be viewed in a GUI via the regedit command:

```
C:\> regedit
```

Or, at the command line, their settings can be observed by running the reg command as follows:

```
C:\> reg query [reg key]
```

The reg command is case insensitive, which is nice if you have trouble remembering the capitalization of the various elements in the registry. Here is a command that reads settings of the Run registry key associated with the local machine (HKLM):

```
C:\> reg query hklm\software\microsoft\windows\currentversion\run
```

© 2016 Ed Skoudis, John Strand

Extra Startup Items

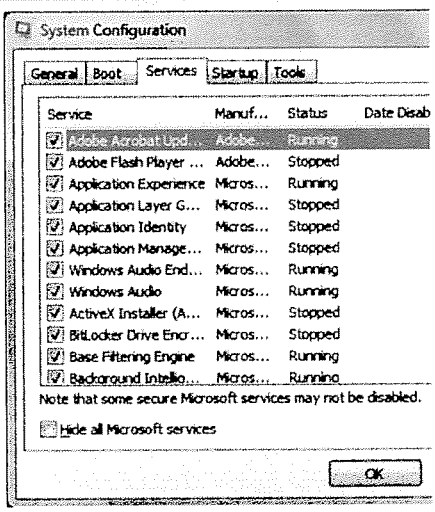
Windows Cheat Sheet

- It's helpful to check users' autostart folders

```
C:\> dir /s /b "C:\Documents and Settings\[user_name]\Start Menu\"
C:\> dir /s /b "C:\Users\[user_name]\Start Menu\"
```

- An administrator can also run msconfig to see what is scheduled to run at startup:
  - Windows→Run... type msconfig.exe
  - Note that msconfig is NOT in cmd.exe's PATH
- WMIC can also show some autostart programs via:

```
C:\> wmic startup list full
```



The cheat sheets also list some of the autostart folders associated with users. These programs are automatically invoked each time the given user logs on to the system, and are sometimes altered by malware.

Various autostart folders and registry key settings can also be analyzed by running a built-in program called msconfig.exe. It's important to note that this program is not in the default PATH of cmd.exe, but it is in the PATH of the Windows button's Run... box. Therefore, you can go to Windows->Run... and type `msconfig.exe` to view it.

A final command that can be used to inspect autostart programs is WMIC, used with the "startup" notation:

```
C:\> wmic startup list full
```


© 2016 Ed Skoudis, John Strand

Unusual Accounts

Windows Cheat Sheet

- Look for new, unexpected accounts in the Administrators group
  `C:\> lusrmgr.msc`
  - Click Groups
  - Double-click Administrators
- At the command line, a list of users is available via
  `C:\> net user`
- A list of members of the Admin group can be seen with
  `C:\> net localgroup administrators`

The Windows cheat sheet asks administrators to look at the users and groups defined on the machine using the "lusrmgr.msc" control. This local user manager interface can be used to check for unexpected accounts in the Administrators group.

At the command line, a list of users can be displayed by running the "net user" command. To see which accounts are in the administrator's group, the following command can be run:

```
C:\> net localgroup administrators
```

© 2016 Ed Skoudis, John Strand

Unusual Files

Windows Cheat Sheet

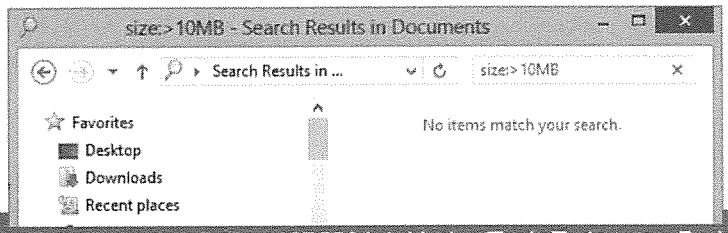
- Check file-space usage for sudden major decreases in space
  - Use GUI (right-click on partition), or type

```
C:\> dir c:\
```

- Look for files larger than 10 MB

```
C:\> FOR /R C:\ %i in (*) do @if %~zi gtr 10000000 echo %i %~zi
```

  - At the GUI
    - On WinXP/2003: Start→Search→For Files of Folders... Search Options→Size→At Least 10000 KB
    - On WinVista through Win8: Explorer search box, then type "size:>10M"



Next, the Windows cheat sheet describes how to look for major decreases in free space on a machine. By using the GUI or the dir command, an administrator can spot normal disk utilization and look for deviations.

We also have them search for unusually large files by using the system search routine to look for files larger than 10 MB. Such files could contain an attacker's sniffer logs, stolen software, or pornography.

This search can be accomplished at a cmd.exe prompt by running

```
C:\> FOR /R C:\ %i in (*) do @if %~zi gtr 10000000 echo %i %~zi
```

This command is based on a FOR loop, which iterates over a set of something. The /R indicates that we are to iterate over files, recursively going through the file system. We start at C:\. Our iterator variable (%i) takes on the different names of various files. We'll iterate over files of any type (in (*)). For each file we iterate over, in our "do" clause, we turn off display of commands (@) and use an IF statement. The iterator variable's file size is referred to as %~zi. We check to see if the file's size is greater than 10 MB (gtr 10000000). If it is, we display (echo) the file's name (%i) and its size (%~zi).

Alternatively, if we have GUI access of the machine, we can look for files with that size using built-in Windows search features.

© 2016 Ed Skoudis, John Strand

Unusual Scheduled Tasks | Windows Cheat Sheet

- Check the scheduled tasks using the Task Scheduler GUI
  - Start→Programs→Accessories→System Tools→Scheduled Tasks
- Look at scheduled tasks on the local host by running

  `C:\> schtasks`
  - Look for unusual scheduled tasks, especially those that run as a user in the Administrator's group, as SYSTEM, or with a blank user name
  - The "at" command shows only those tasks scheduled using the "at" command... not those scheduled with schtasks
  - The schtasks command shows those scheduled with "at" and with schtasks

Administrators also need a good feel for what tasks are normally scheduled to run on their systems. The Task Scheduler GUI and the schtasks command can be used to list scheduled tasks. The Windows cheat sheet requests that system administrators look for unexpected tasks that are scheduled to run with Administrator, SYSTEM, or blank privileges, which might be signs of an attack.

An older feature, the "at" command, can also be used to create tasks and display them, but it is limited. The "at" command only displays those tasks created using the "at" command itself, and not those scheduled tasks created with schtasks. The schtasks command show a more comprehensive set of scheduled tasks, displaying those tasks created using schtasks itself, the task scheduler GUI, and the "at" command.

© 2016 Ed Skoudis, John Strand

Unusual Log Entries

Windows Cheat Sheet

- To look at logs, run the Event Viewer
  `C:\> eventvwr.msc`
- Look for suspicious events
  - "Event log service was stopped."
  - "Windows File Protection is not active on this system."
  - "The MS Telnet Service has started successfully."
  - Look for a large number of failed logon attempts or locked-out accounts
- Via the command prompt, on some versions of Windows, an admin can inspect logs with
  `C:\> eventquery.vbs /L security`
  - Unfortunately, Windows Vista, 7, and 8 do not include the "eventquery.vbs" command; instead, you could use
  `C:\> wevtutil qe security /f:text`

By running the Event Viewer control (eventvwr.msc), an administrator can look for anomalous event logs. Some of the most telling events to look for include

- An indication that the event log service was stopped, which may have been done by an attacker to cover tracks.
- A sign that the built-in Windows file integrity checker (Windows File Protection) was disabled.
- A sign that the Microsoft Telnet service has been invoked.
- An indication of a large number of failed logon attempts or locked-out accounts.

From a command-line perspective, some versions of Windows include the eventquery.vbs, which is a Visual Basic script. This tool can view all logs by running it as

```
C:\> eventquery.vbs
```

To narrow down just to security logs, the script can be run with

```
C:\> eventquery.vbs /L security
```

Unfortunately, Windows Vista, 7, and 8 do not include the "eventquery.vbs" command. Instead, on those more recent versions of Windows, you could run

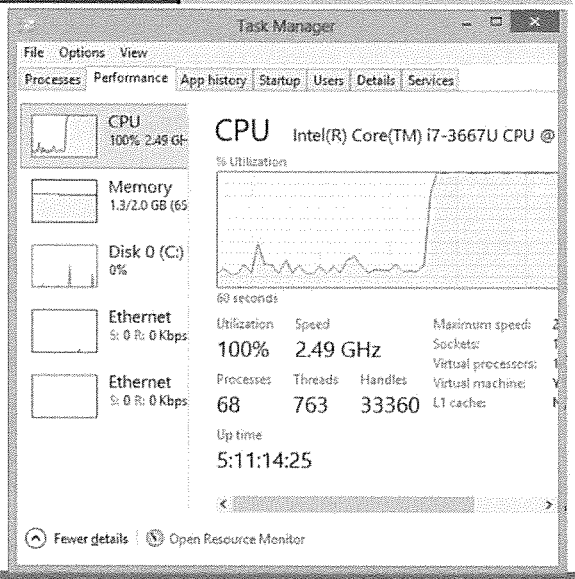
```
C:\> wevtutil qe security /f:text
```


© 2016 Ed Skoudis, John Strand

Other Unusual Items

Windows Cheat Sheet

- The cheat sheets tell administrators to check the performance monitor...
- ...and look for unusual system crashes



As a final element, the cheat sheets remind the system administrator to look at the performance-monitoring tool associated with the Task Manager (Task Manager→Performance Tab) to see how the system is performing. Also, it's recommended that system administrators look for unusual system crashes. Of course, neither circumstance is a guarantee of an attack, but these are items for the admins to keep in mind when analyzing their systems.

© 2016 Ed Skoudis, John Strand

Additional Supporting Tools

Windows Cheat Sheet

- Tools for mapping listening TCP/UDP ports with the program listening on those ports
  - TCPView: Free
- Additional process-analysis tools
  - Process Explorer and Process Monitor, free at https://technet.microsoft.com/en-us/sysinternals/default.aspx
  - Center for Internet Security templates and scoring tools

Beyond the built-in capabilities of Windows, some favorite additional tools for checking the security status of a machine include the TCPView tool, which shows listening TCP and UDP ports, as well as the program name that is listening on those ports.

Two other useful process-analysis tools are Process Explorer and Process Monitor from Microsoft Sysinternals.

The Center for Internet Security also has hardening templates and scoring tools for Windows. They are amazingly useful starting points for hardening Windows systems.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. Incident-Handling Process
3. Preparation
4. Identification
   - **Lab: Windows Cheat Sheet**
5. Containment
6. Eradication
7. Recovery
8. Lessons Learned
   - Lab: Enterprise-Wide Identification and Analysis

Now that we've got a feel for the Windows cheat sheet, let's do some hands-on lab work with Windows. Some of these labs just look at our box in its normal state. In other lab steps, we actually create the condition we want to detect, and then run the cheat sheet tip to detect it.

© 2016 Ed Skoudis, John Strand

Windows Cheat Sheet Lab

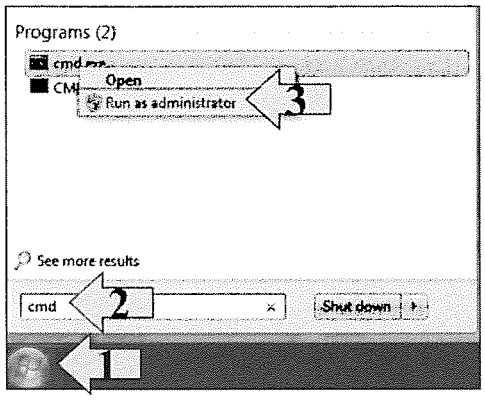
- Let's run through a Windows lab on the cheat sheets
- Using Windows, we explore how the commands in these cheat sheets work
- For some elements, we just run the command
- For others, we create the unusual condition, and then show how the command detects it
- The lab requires any version of Windows from Win XP Pro through Windows 8
  - That's specified in the course requirements online
- If you brought an earlier or less powerful version of Windows, follow along
  - Work with someone near you

The lab associated with this cheat sheet requires Windows XP Pro or later (up to and including Windows 8). Those are the Windows versions that were specified in the requirements when you signed up for the class. If you do not have those versions of Windows, follow the labs on your own machines, realizing that some of the elements will not work due to a lack of features in your operating system. Also, you may want to work with a person sitting near you who has a system compatible with the course Windows requirements. We strongly encourage you to do so.

© 2016 Ed Skoudis, John Strand

Lab: Launch a cmd.exe Shell with Elevated Privileges

- For this lab, you need a shell with full admin privileges
  - On Windows XP or 2003, simply log in with admin privileges and go to Start-->Run... and type cmd.exe
    - Or use C:\> runas /u:[admin] cmd.exe
  - On Windows Vista, 7, or 8, you need an ***elevated command shell***
    - Don't just run a cmd.exe and expect it to have such privileges because you are logged into the GUI as an administrator
    - Click the Windows button, search for cmd, and then right-click cmd.exe and select "Run as administrator"
    - Or type cmd.exe and press CTRL-SHIFT-Enter



For this lab, you need a command shell with full administrator privileges. On Windows XP or 2003, getting such shell access is easy. If you are currently logged into the GUI with admin privileges, simply go to Start-->Run... and type cmd.exe. If you aren't currently logged in as admin, simply launch a non-admin cmd.exe and then use the runas command to launch a cmd.exe via the command "runas /u:[AdminUser] cmd.exe."

Special Note for Windows Vista, 7, and 8 Users

If you are using Windows Vista or later, when you simply invoke a cmd.exe via the GUI, you won't have full administrator privileges, even if you are logged into the GUI as an administrative account. That's because Windows is trying to protect your system from you, not giving you full administrator privileges at the command line unless you specifically demand it. For many of the commands you are going to run in this lab, you need an **elevated command shell**, which has full admin privileges. To get such shell access, go to your Windows icon (Step 1 in the slide above). Do a search for cmd (Step 2). And, in Step 3, right-click cmd.exe, select "Run as administrator." Now you have an elevated command shell, which you can use for all the labs in this class that require admin access.

© 2016 Ed Skoudis, John Strand

Lab: Network Usage – Netstat

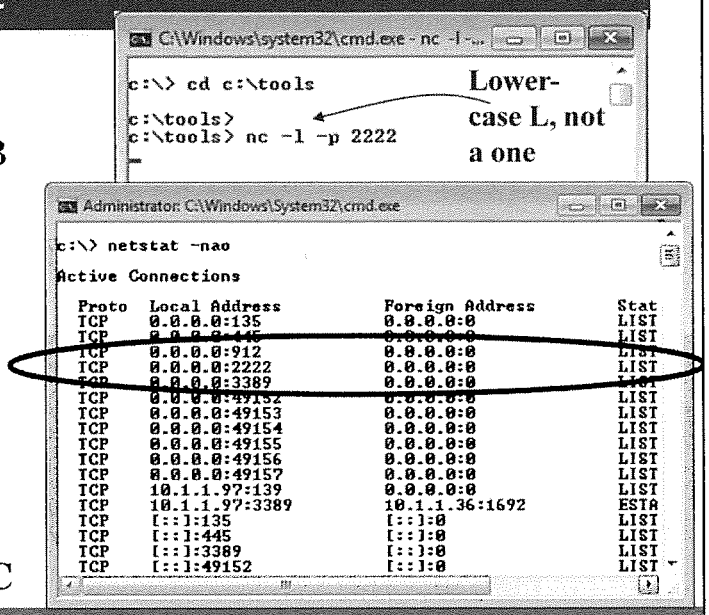
- Let's create a Netcat backdoor listener and look for its open port
- Unzip Netcat from the Course USB into c:\tools
- From a command prompt, run Netcat

```
C:\> nc -l -p 2222
```

- At a separate command prompt, run netstat

```
C:\> netstat -nao
```

- Add a 5 to the end of the netstat command
- Stop Netcat and netstat by CTRL-C



Let's look for unusual TCP and UDP ports with the netstat command.

First, unzip the Windows\Netcat.zip file from the USB, placing nc.exe in the directory c:\tools on your hard drive. Netcat is a tool that can use TCP and UDP ports to send or receive data. We'll be doing a lot with Netcat on Day 3 of this course. After unzipping Netcat, create a Netcat listener on a TCP port by typing:

```
C:\> cd c:\tools
C:\> nc -l -p 2222      (That –l is a dash-lower-case-L, not a dash-one.)
```

Your personal firewall may block these connections. Temporarily allow them so we can run the lab. Now, in another window, look for that listener by running:

```
C:\> netstat -na
```

Try running this also, to see the process ID numbers for the executables using the TCP and UDP ports:

```
C:\> netstat -nao
```

That PID sure is helpful, especially when cross-referenced with the output of Task Manager, the tasklist command, or the wmic process output.

Finally, try running this, which will give an updated netstat output every 5 seconds:

```
C:\> netstat -na 5
```

Next, try running netstat with the –naob flag to see the EXE and DLLs associated with each listening port.

```
C:\> netstat -naob
```

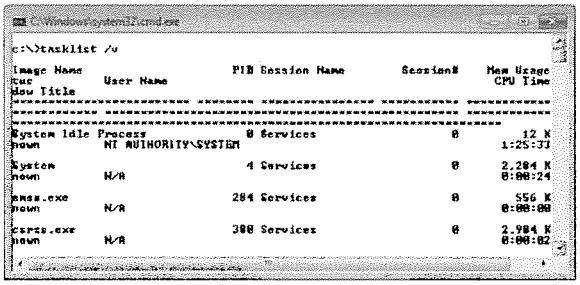
On Windows Vista, 7, or 8, to run netstat with the -b flag, you need a shell with elevated privileges (not just admin privileges, but elevated privileges, which give more access to the underlying operating system). To get an elevated privilege prompt, click on your Start--> menu, and type cmd.exe in the search box. DO NOT HIT ENTER YET. When the cmd.exe icon appears, right click on it, and select "Run as administrator." The UAC feature may prompt you to confirm running cmd.exe with higher privileges. Select "Yes." Then, run netstat -naob.

Finally, stop your Netcat listener by hitting CTRL+C in its window.

© 2016 Ed Skoudis, John Strand

Lab: Unusual Processes

- Check which processes are running on your box by using three methods
  - Task Manager GUI
    `C:\> taskmgr.exe`
  - Tasklist command
    `C:\> tasklist /v`
  - WMIC command
    `C:\> wmic process list full`



- Note the differences in the information shown in the output

First, we look at the processes running on our machines, in several ways. The most common way to look at running processes is to invoke the Task Manager, by running taskmgr.exe:

```
C:\> taskmgr.exe
```

Look at the Process tab. If your Task Manager does not show the Process ID numbers (the default on most Windows machines does not show this highly valuable information), reconfigure Task Manager to do so. In Task Manager, go to View→Select Columns. Check the PID (Process Identifier) check box, and click OK.

Next, look at the running processes by invoking the tasklist command, in verbose mode. This shows us the user that each process is running as, among other useful information:

```
C:\> tasklist /v
```

Finally, to get more details associated with each running process, use the wmic command:

```
C:\> wmic process list full
```

On many Windows machines, the first time you execute wmic, *you'll see a note saying that* wmic *is being installed. The install package is already included in the operating system, but you have to run the command one time to do the installation. The installation should take less than a minute, and then your system will be able to use the* wmic *command.*

Note the different information you can get for each process with these different commands. What interesting information do you see in the wmic output that you cannot see in the other output? One item that is useful is the command line used to invoke each process, as well as its ParentProcessID. Do you see any other differences that might make one command more useful than the others for given kinds of analysis?

© 2016 Ed Skoudis, John Strand

Lab: Unusual Services

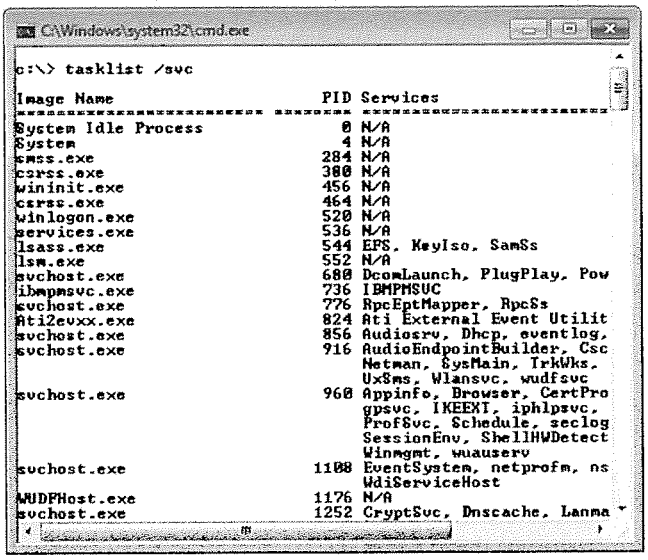
- Look at the services running on your system

```
C:\> services.msc
C:\> sc query | more
```

- Next, look at process-service mapping

```
C:\> tasklist /svc
```

- What process is the RPC Service running out of?



Now, let's look for the services that are defined and started on the box. At a command prompt, run

```
C:\> services.msc
```

That'll bring up the services control panel. Also, run the sc command to get more details, piping its output through the more command to display it one page at a time:

```
C:\> sc query | more
```

Now, let's look at how various services map to the processes they are running out of. The tasklist command shows this mapping with the following command:

```
C:\> tasklist /svc
```

As an example of a useful form of analysis, what process is the RPC Service (known as RpcSs) running out of? That is one busy process on most Windows machines, running several different services simultaneously.

Again, your system admins should have a pretty good feel for what services are running and why.

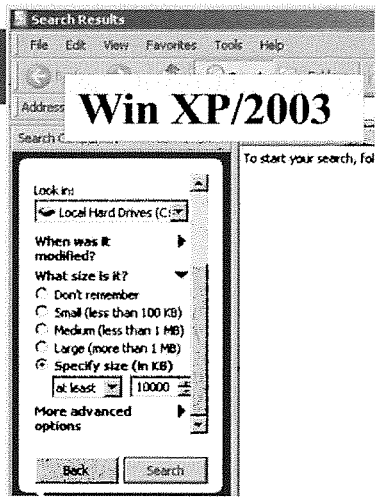
© 2016 Ed Skoudis, John Strand

Lab: Unusual Files

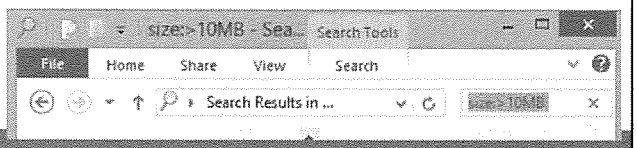
- Look for files larger than 10 MB
  - At the command line

```
C:\> FOR /R C:\ %i in (*)
  do @if %~zi gtr 10000000
  echo %i %~zi
```

  - At the GUI
    - On WinXP/2003: Start→Search→For Files of Folders... Search Options→Size→At Least 10000 KB
    - On WinVista, Win7, and 2008: Launch File Explorer and then search for "size:>10M"



Win 7/8



Now, let's look for unusual files. A common approach is to look for files larger than 10 MB, although, for some systems used for video or audio editing, you may need to increase that size to find outliers of larger size. You can do this at the command line by running:

```
C:\> FOR /R C:\ %i in (*) do @if %~zi gtr 10000000 echo %i %~zi
```

This command is based on a FOR loop, which iterates over a set of something. The /R indicates that we are to iterate over files, recursively going through the file system. We start at C:\. Our iterator variable (%i) will take on the different names of various files. We'll iterate over files of any type (in (*)). For each file we iterate over, in our "do" clause, we turn off display of commands (@), and use an IF statement. The iterator variable's file size is referred to as %~zi. We check to see if the file's size is greater than 10 MB (gtr 10000000). If it is, we display (echo) the file's name (%i) and its size (%~zi).

Alternatively, if we have GUI access of the machine, we can look for files with that size using built-in Windows search features.

On Windows XP or 2003, go to Start-->Search-->For Files or Folders. Select "All Files and Folders," "What size is it?," "Specify Size," "at least," and 10000KB. Then, run it by clicking "Search Now" or "Search."

On Windows Vista, 7, and 8, go to Start--> and type "size:>10M." Then, hit Enter. Note that this search can alternatively be based on kilobytes (K) or Gigabytes (G).

What did it find?

If you lower it to about 1.5 Megs, you might find some media files. Any songs? How about video? Be careful here....

© 2016 Ed Skoudis, John Strand

Lab: Unusual Registry Key Settings

- Let's look for unusual startup items kicked off by the Run, RunOnce, and RunOnceEx registry keys
- Using the regedit command, look in
  - HKLM\Software\Microsoft\Windows\CurrentVersion\Run
  - HKLM\Software\Microsoft\Windows\CurrentVersion\Runonce
  - HKLM\Software\Microsoft\Windows\CurrentVersion\RunonceEx
- Using the reg command, look at those locations

  `C:\> reg query [key]`
- You may or may not have entries for Runonce or RunOnceEx
- Finally, check out autostart entries by going to Windows Run and typing "msconfig.exe"
  - Look at the Startup tab

Now, we'll look at programs that are automatically invoked by our machines when the system boots up or when a given user logs on. In particular, we'll look at the Run, RunOnce, and RunOnceEx registry keys.

Invoke the graphical tool for analyzing the registry, Regedit, and navigate to each of these keys to see their settings:

```
C:\> regedit
```

Look at

HKLM\Software\Microsoft\Windows\CurrentVersion\Run

HKLM\Software\Microsoft\Windows\CurrentVersion\Runonce

HKLM\Software\Microsoft\Windows\CurrentVersion\RunonceEx

By the way, HKLM stands for "Hkey_local_machine."

Next, check out each of these key settings using the reg command at the command line, as in

```
C:\> reg query hklm\software\microsoft\windows\currentversion\run
```

Note that you can hit the "up" arrow and append Once and OnceEx to this command to ease the viewing of these keys. Also, prepend HKCU in front to look at the autostart reg keys associated with your currently logged on user. Note that you might not have any autostart programs for RunOnce and RunOnceEx defined on your machine.

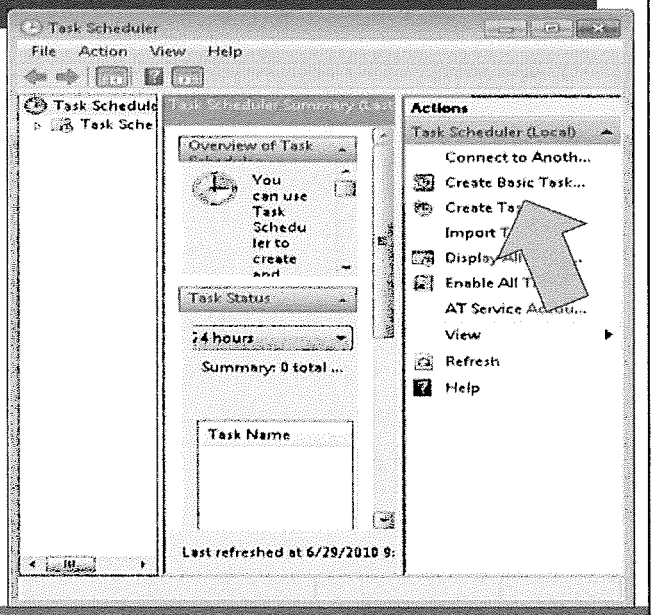
Finally, perform a similar analysis by invoking msconfig.exe and viewing the Startup tab:

Start > Run … and type "msconfig.exe."

© 2016 Ed Skoudis, John Strand

Lab: Unusual Scheduled Tasks

- In XP and 2003 Start→Programs→Accessories→ System Tools→Scheduled Tasks
  - Right-click the Scheduled Tasks window, go to New→Scheduled Task
  - Name your task "test"
  - Double-click it, and have it "Run:" the ipconfig.exe command; click OK
- In Windows Vista, 7, or 8
  - Control Panel→Administrative Tools
  - Launch Task Scheduler and select Create Basic Task
  - Step through wizard with task name "Test," description "Test," run "One time," any date, action of "Start a program," and a program of "ipconfig.exe"

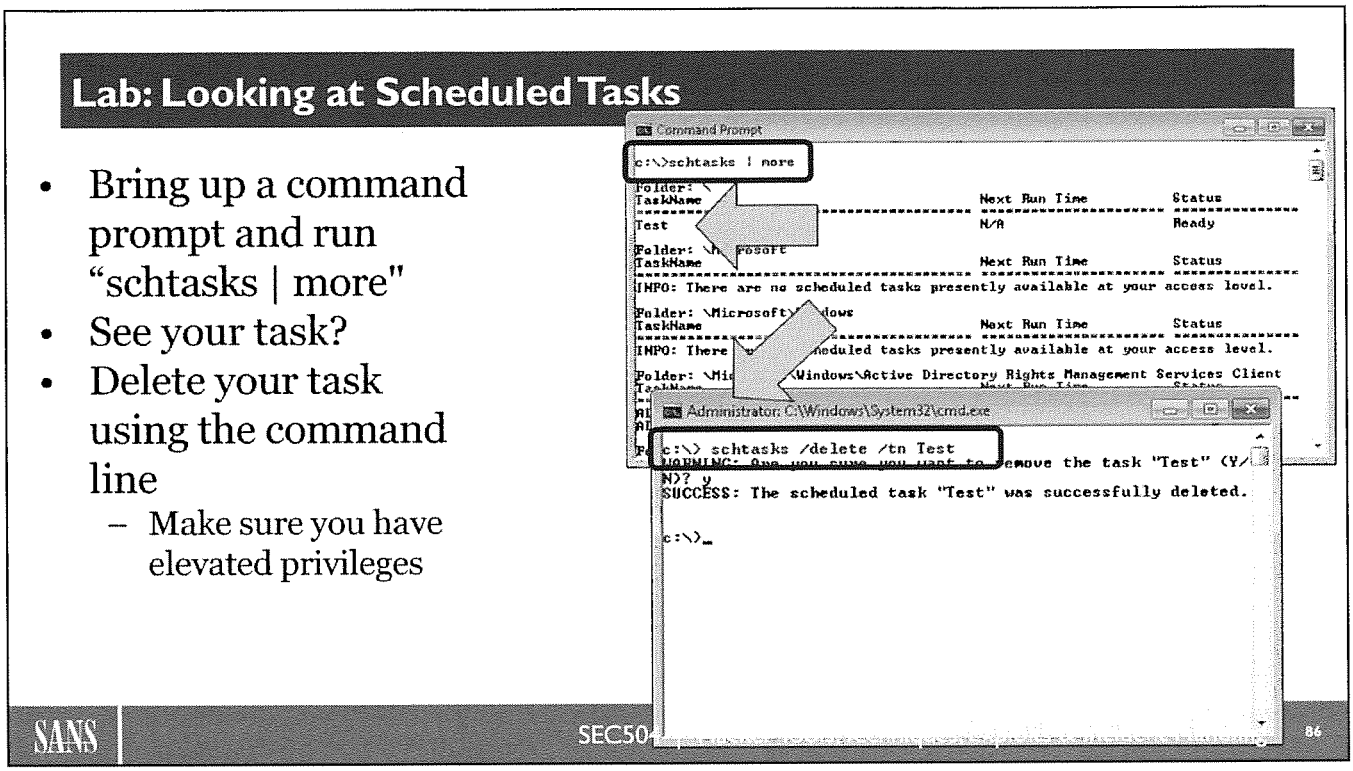


To look for unusual Scheduled Tasks, we'll invoke the Task Scheduler:

- On Windows XP or 2003, select Start→Programs→Accessories→System Tools→Scheduled Tasks. Now, schedule a task by right-clicking in the Scheduled Tasks window, and going to "New→Scheduled Task." Name your task "test." Double-click the task, and have it "Run:" the ipconfig.exe command by simply typing ipconfig in the "Run:" box. Click "OK." You may have to type a password with which the task to run. You should see the new task in your Scheduled Tasks window.

- On Windows Vista, 7, or 8, go to your Control Panel, select View By "Small Icon," and select "Administrative Tools." Double-click the Task Scheduler when it appears. Then, in the Task Scheduler GUI, on the right-hand side, select "Create Basic Task…." Walk through the wizard and create a task named "Test" with a description of "Test." Click Next. Now, set a schedule to run "One time" at any date and time of your choosing. The action should be "Start a program," and the program is "ipconfig.exe." In this example, no command-line arguments are needed for ipconfig.

© 2016 Ed Skoudis, John Strand



Now, look for your scheduled task. Let's try the "schtasks" command. Run it as follows:

```
C:\> schtasks | more
```

Do you see your task? Depending on the version of Windows, it should be near the top or bottom of the list.

Besides your Test task, do you know what the other tasks defined on your system actually do?

Finally, delete your task by running the following command:

```
C:\> schtasks /delete /tn Test
```

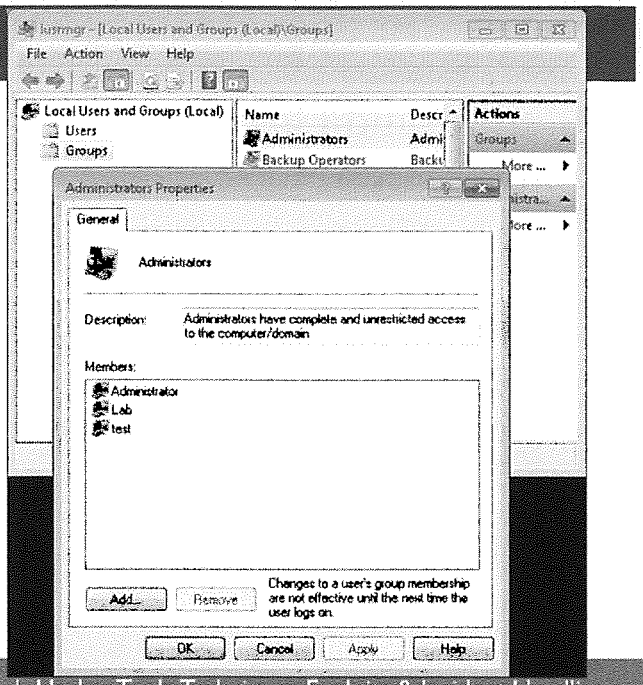
If a message says "Access is denied," you need to invoke a command shell with elevated privileges. Do so by going to the Start--> menu and searching for cmd.exe. Do not hit Enter. Instead, right-click cmd.exe and select "Run as administrator." The resulting cmd.exe should include "Administrator:" in its window title bar, indicating that you have elevated privileges for this prompt. Now, go back and delete the task.

© 2016 Ed Skoudis, John Strand

Lab: Unusual Accounts

- At a command prompt, run lusrmgr.msc
- Look at the users on the box and the groups
- Who is in the Administrators group?
- Check this information by running:

```
C:\> net localgroup
administrators
```



Let's look at the accounts on our box. You can bring up the user manager console in a variety of ways. The easiest is just by typing this at a command prompt:

```
C:\> lusrmgr.msc
```

Now, look at the users defined on your box. Also look at the groups. Double-click the Administrators group. Who has those privileges on your machine? Do you know all of those people?

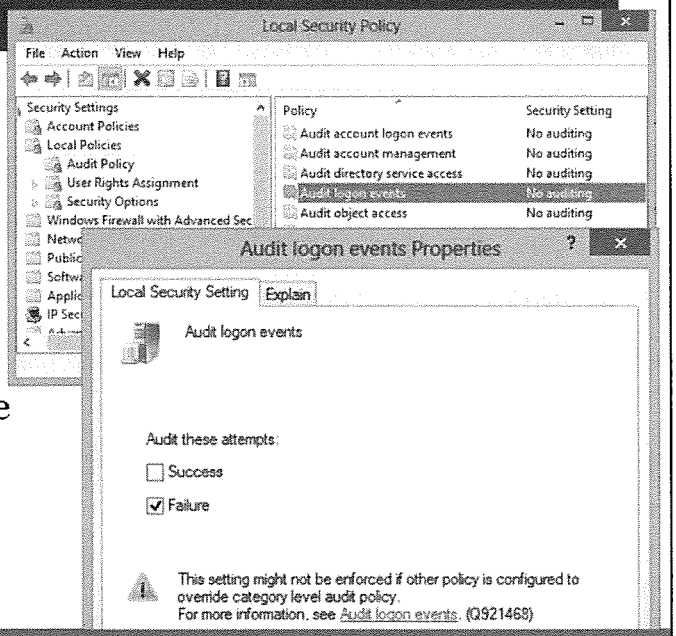
For another view of the users defined on your system that are in the Administrators group, run the following command:

```
C:\> net localgroup administrators
```

© 2016 Ed Skoudis, John Strand

Lab: Unusual Log Entries

- Alter the audit log configuration
  - Go to Windows launch menu and type
    `secpol.msc`
  - Note that on some systems, secpol.msc is not in the cmd.exe PATH
  - Go to Local Policies→Audit Policy→Audit Logon Events
  - Select "Failure"



Our final element of this lab is to look at unusual log entries. To set this up, you need to go to your audit log configuration.

Bring up your local security policy editor by running

```
C:\> secpol.msc
```

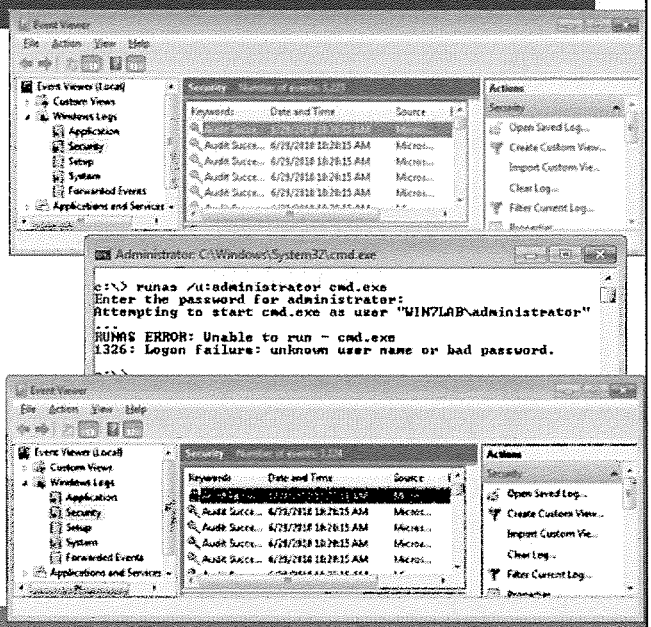
Now, go to Local Policies→Audit Policy. Double-click Audit Logon Events. Select "Failure."

Our box is now configured to log whenever someone tries to log on to the box and doesn't provide the proper password. Amazingly, that's not on by default.

© 2016 Ed Skoudis, John Strand

Lab: Generating a Security Event

- Run the Event Viewer (eventvwr.msc)
- Look at "Security Log" or "Windows Logs" and "Security"
- Now, generate an event by running: "runas /user:Administrator cmd.exe"
- When it asks for a password, type in something bogus
- Look at your Event Viewer and Hit Refresh (F5 key)



Next, bring up your Event Viewer. The easiest way to do that is to run at the command prompt:

```
C:\> eventvwr.msc
```

Look at your Security Log in the Event Viewer.

Let's generate an event by using the runas command at the command prompt:

```
C:\> runas /user:Administrator cmd.exe
```

When it asks for a password, type in something bogus (not your real password!).

Now, hit Refresh (by hitting the F5 Function key) in the Event Viewer. See your event? We all want our sysadmins to be on the lookout for that kind of event, provided that our boxes are configured to generate them in the first place.

© 2016 Ed Skoudis, John Strand

Going Further

- There is a challenge .exe script on the class USB in the Windows Directory
  - 504lab.exe
  - Copy the correct .exe to your C:\Tools directory
- When you run the script, it asks you a series of random questions on ports, processes, and command-line invocations
  - Every time will be different
- Generously provided by SANS Instructor Mark Baggett
- Please run it as Administrator!

We have added a cool script to practice your Windows command line kung-fu.

Please copy these from the class USB in the Windows directory to your tools directory and run them from the command line. Feel free to run them multiple times, as the answers will change every time.

© 2016 Ed Skoudis, John Strand

Running the Script

```
KNOW THY SYSTEM!

Open a second CMD prompt as an Administrator and run netstat -nao on your host so you know what your system loo
ks like before it is "infected."
Verify your firewall and AV are disabled.  I am about to start a non-malicious backdoor for you to find.

After you have run netstat press ENTER to continue


Please wait: A TCP Backdoor is being started on your host.
Backdoor Started.  Please answer the following questions.

What TCP port is the backdoor listening on? 49883

What is the process id number of the backdoor? 4676

What is the Parent process id number of the backdoor? _
```

- Run it as Administrator!
- Disable your firewall!
  - `C:\>netsh advfirewall set allprofiles state off`

Above are the instructions for running the script. Remember the answers change with each run!

Just a few notes:

1. Be sure to run it as administrator
2. Disable your firewall `netsh advfirewall set allprofiles state off`
3. Some answers may require you to dig a bit. Look through the slides and remember! Google is your friend!

© 2016 Ed Skoudis, John Strand

Lab Conclusions: Ideas for Applying Cheat Sheets

- To successfully use the cheat sheets, you need a communications plan
  - Don't just throw them over a wall
- Print out one for each system admin and distribute
  - Laminate(?)
- Distribute to Operations management
- Weekly/monthly Sys Administration meetings
  - Presentation covering use of the cheat sheets
- Periodic conference call
  - Best practices
- Separate calls for Windows and Linux administrators
  - Otherwise, they might get bored or turn it into a war!

There we have it … we looked at our Windows cheat sheets. You can learn more about the Linux cheat sheet in the appendix. But, how can you apply either cheat sheet in your organization?

You simply cannot throw them over the wall and expect your system administrators to integrate them into their processes. You'll need to help them do that. First, you can print them out and distribute them to administrators and their managers.

Then, try to get an invite to a weekly or monthly system administrators meeting or conference call in your organization. Most Sys Admin teams meet on a periodic basis to discuss events and techniques. During such a meeting, you could describe the cheat sheets to the admins and ask them to use them.

Also, if your organization offers system-administrator orientation or refresher training, incorporate the cheat sheets into those materials.

I caution you about simultaneously presenting both cheat sheets. You might want to run through the Linux sheet on one call or meeting, and then do the Windows sheet during the next meeting. Otherwise, you may run into a religious war, or risk boring some aspects of your Sys Admin team.

Finally, the only way we can get any value out of the cheat sheets is by exposing them to our system administrators.

© 2016 Ed Skoudis, John Strand

Initial Identification Assessment

- Determine whether an event is actually an incident
  - Check for simple mistakes by users, admins, or others
  - Assess the evidence in detail
  - Ask yourself, "what other possibilities are there?"
  - We'll spend the rest of our class discussing the methodology used by attackers to help familiarize you with items to watch for
  - Maintain situational awareness, reporting to chief

Efficient handling of errors is part of the process

A large number of "incidents" turn out to be false positives, or perhaps a nicer term is events. One of the most important functions of an incident handler is to continue to assess the data to see if it indicates this could be something other than an incident. The process should be optimized to encourage reporting and then deal with false positives gracefully and efficiently. Keep in mind that this can be great training, especially for the second, or junior person sent to the scene. A wise handler, who can already see what the situation is, can use the opportunity to help educate the less experienced handler.

One of the most important responsibilities of a senior incident handler is to maintain situational awareness. What is the effect of the vulnerability, can it be remotely exploited, is an exploit available, or is this a zero-day attack (an attack that was previously unannounced)? These types of questions help the handler come to a reasonable initial assessment.

© 2016 Ed Skoudis, John Strand

Assessment Questions

- When looking at the situation, you need to determine how much damage could be caused:
  - How widely deployed is the affected platform or application?
  - What is the effect of vulnerability exploitation, if a vulnerability is present?
  - What is the value of the systems impacted so far? What is the value of the data on those systems?
  - Can the vulnerability be exploited remotely (via a network connection)?
  - Is a public exploit available? Was one recently released?

Purists may choose to use a point system, but the main idea is that, by asking the right questions, you can come to a reasonable initial assessment.

How widely deployed is the affected platform or application? Obviously, the more widely deployed, the greater the risk. A custom application means the containment is simple. A vulnerability affecting multiple Windows platforms is really scary.

In terms of the effect, is it denial of service, reconnaissance, large-scale reconnaissance, information compromise, user compromise, privileged user (root or administrator) compromise?

What is the value of the systems impacted so far? What is the value of the data on those systems? Obviously, high-value victim machines, or systems storing sensitive data represent a much more significant risk.

Can the vulnerability be exploited remotely (via a network connection)? If not, again containment is reasonable, if by the internal LAN, again containment is not too tough, if by the Internet you may need to configure a firewall or router rule fast.

Is a public exploit available for the vulnerability? One source to check is the Common Vulnerabilities and Exposures web page at cve.mitre.org, also bugtraq and isc.sans.edu. If there is no mention of a public exploit you may be dealing with a zero day (not previously announced) vulnerability and exploit. This could be evidence of a high end attacker and raises the stakes.

© 2016 Ed Skoudis, John Strand

Additional Assessment Questions

- Ask yourself:
  - What level of skill and prerequisites are required by an attacker to exploit the vulnerability?
  - Is the vulnerability present in a default configuration?
  - Is a fix available for the vulnerability?
  - Do other factors exist that reduce or increase the vulnerability's risk or potential impact, such as the possibility it is a worm?
- Lenny Zeltser has prepared an Initial Security Incident Questionnaire for Responders
  - http://zeltser.com/network-os-security/security-incident-questionnaire-cheat-sheet.html

What level of skill and prerequisites are required by an attacker to exploit the vulnerability? Also, does he have the skills to do anything to the system after he breaks in?

Is the vulnerability present in a default configuration? All too often, the answer to this is yes. The chances you are running default configurations are high, so this raises the risk level.

Is a fix available for the vulnerability? If so, this will be a major part of containment.

Do other factors exist that reduce or increase the vulnerability's risk or potential impact, such as the possibility that it is a worm? This can be the reason you drop your Internet connection for your entire organization. In the case of Nimda, UUNET estimates it reached saturation in about two hours across the entire Internet. For SQL Slammer, a massive outbreak occurred in 15 minutes.

Lenny Zeltser, SANS Instructor, has written a cheat sheet with these and more questions on it to ask while responding to security incidents. Lenny's cheat sheet is available at http://zeltser.com/network-os-security/security-incident-questionnaire-cheat-sheet.html.

© 2016 Ed Skoudis, John Strand

Identification – Establish Chain of Custody

- Be careful to maintain a provable chain of custody
  - Do NOT delete ANY files until the case is closed out, and even then if you have storage space, save them for a document retention timeframe approved by your legal team
  - Identify every piece of evidence in your notebook
  - Control access to evidence
- Each piece of evidence must be under the control of one identified person at all times
  - Include a lined page with the evidence to record all hand-offs: who and when
  - Record when you lock it up in storage
- When turning over evidence to law enforcement, have them sign for it

A recommended practice is to keep everything together for a particular case. Before you touch anything, if there is reason to suspect this could go to court, it is wise to fill out attestation forms to the tune of "I, John Doe, 1 April 2013, am in room 23, 1416 Able St, and am looking at a Dell server, serial number XXX. This computer is suspected of being involved in criminal activity. At 21:45, we are disconnecting the network cord. We have done nothing else with this machine." And on it goes. To the extent possible, account for every action you take or command you type. Cameras, if allowed, can be useful. Often, all the evidence will fit in a gallon-size Ziploc bag, and you can read the evidence listing through the plastic. Keep it in a locked container that very few people can access.

To help maintain an inventory of all evidence you exchange with law enforcement and improve the chain of custody you maintain, have law enforcement officials sign for all evidence you hand over; make sure all the evidence is accounted for, and make sure the list includes some description of the "value" of the evidence. Also, give them a copy of your evidence, not the originals, unless they specifically require and ask for the originals. Most of the time, copies will suffice.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. Incident-Handling Process
3. Preparation
4. Identification
   - Lab: Windows Cheat Sheet
5. **Containment**
6. Eradication
7. Recovery
8. Lessons Learned
   - Lab: Enterprise-Wide Identification and Analysis

Let's review where we are in the process. We have prepared to some extent. We have identified a possible incident. We have gone on scene and we have introduced chain of custody.

The goal of containment is to keep the problem from getting worse. Before we fire, we should take the time to aim! Try to do a decent survey and review of the situation before altering the system.

When an incident handler first arrives on location, there is a chance that the system is pristine in terms of evidence and information. As soon as the handler starts to recover the system, there is a point in which the evidence starts to become contaminated. If at all possible, the file-system image creation should come before this point so there is a copy of the unaltered system.

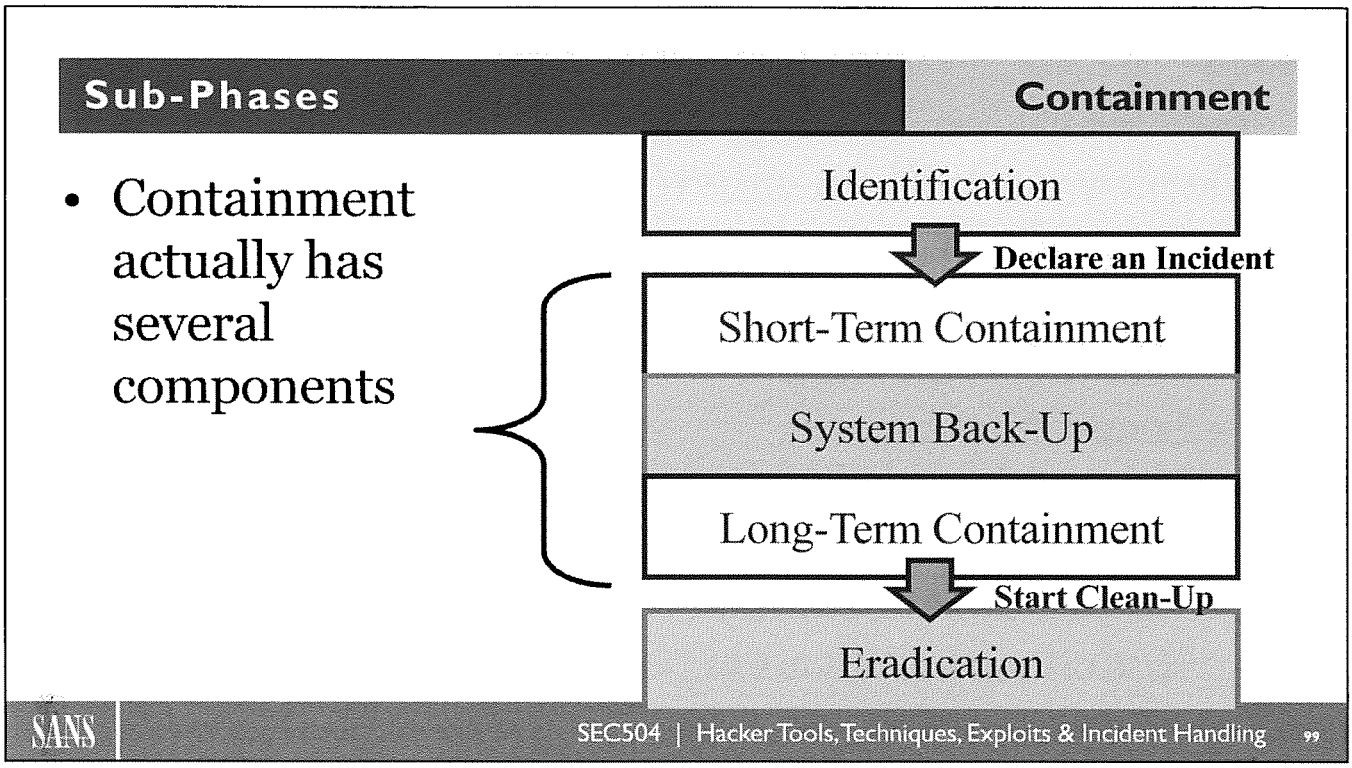
© 2016 Ed Skoudis, John Strand

Containment

- The goal of the Containment phase is to stop the bleeding
  - Prevent the attacker from getting any deeper into the impacted systems or spreading to other systems
- We discuss
  - The sub-phases of Containment
  - Methods for short-term Containment
  - System back-up
  - Methods for long-term Containment

For Containment, we want to stop the bleeding. How can we arrest the attacker in his/her tracks before he/she causes more damage? That's what Containment is all about!

© 2016 Ed Skoudis, John Strand



Containment includes three sub-phases: short-term containment just to stop the damage, followed by system back-up, followed by long-term containment to make sure the bad guy is denied access. Let's look at each of these in more detail.

© 2016 Ed Skoudis, John Strand

Deployment | Containment

- Deploy a small on-site team to survey the situation
  - Typically, these will be the same personnel as the Identification team
  - Secure the area
  - If possible, use preprinted survey forms provided at www.sans.org/score/incidentforms
  - Review the information that was provided to you from the Identification phase

If you are dealing with a suspected crime, still or digital cameras can be used to record the scene, where you were, where things were located and who was in the room.

A recommended practice during a survey is to trace down all the wires in a room; be especially alert for impromptu networks and anything that is telephony or wireless.

The incident actually started before you arrived; don't treat time zero as your arrival on the scene! What you see when you arrive may not be what the original user saw; things could change. For example, the RingZero Trojan would start on the desktop, but then remove itself from the desktop. If the user says he saw an icon on the desktop, and you looked and didn't see it and decided he was wrong, this doesn't make you a world-class handler, now does it? Take the time to review the evidence that was created before you arrived on the scene. This includes what everyone saw, heard, and did. It also includes whatever documentation was made.

© 2016 Ed Skoudis, John Strand

Characterize Incident

Containment

- Given that we have declared an incident, we need to record its category (one or more), severity (based on current understanding... subject to change), and sensitivity
- Category
  - Denial of Service
  - Compromised Information
  - Compromised Asset
  - Unlawful Activity
  - Internal Hacking
  - External Hacking
  - Malware
  - E-mail
  - Policy Violations
- Criticality
  - 1) Incident impacts critical systems: 60 min
  - 2) Incident impacts non-critical systems: 4 hrs
  - 3) Possible incident, non-critical: 24 hrs
- Sensitivity: Who should be informed?
  - 1) Extremely sensitive (CSIRT, mgmt)
  - 2) Sensitive (CSIRT, mgmt, sys owners, ops)
  - 3) Less sensitive (employees informed of isolated virus infection)
- Detailed CSIRT Case Classification document available at http://www.first.org/_assets/resources/guides/csirt_case_classification.html

Sample Response Times for you to customize

In moving to the Containment phase, we have declared an incident. It is important to document various characteristics of the incident early on in our Containment phase. The FIRST organization distributes an incident Case Classification document that recommends characterizing an incident based on three areas: it's general category, the criticality of impacted systems and data, and the sensitivity with which information about the case itself should be treated.

From a category perspective, most incidents fall into one or more areas of the list shown above. It is important to note that a single incident may be in multiple categories, such as Compromised Information, Malware, External Hacking, and E-mail, all in the same incident. You may want to add new categories here as attacks evolve.

The criticality rating of an incident will help determine how quickly you'll need to assign a team and deploy to handle the situation. For highly critical incidents, you may want to establish a baseline of response time at 60 minutes, or perhaps even less for some organizations with critical computing needs. Customize these timeframes based on the type of information your organization handles and the criticality of its computing base.

The sensitivity metric here determine the types of personnel with whom information about the incident can be shared. For a case that is extremely sensitive, we may only want to share information with the incident response team and management. For sensitive cases, we may add in the system owners and the operations teams. For less sensitive cases, we may inform more employees, such as in the case of an isolated virus infection.

© 2016 Ed Skoudis, John Strand

Inform Management — Containment

- Identify a senior management sponsor for your team
- CISO, CIO, Legal Counsel, etc.
- When you declare an incident, notify your management sponsor and get help to assist in the incident-handling process
  - Notification may be a mere e-mail...
  - For more serious incidents, a phone call or visit
- Get a copy of the corporate phonebook
- Assign a minimum of two people to each incident: a primary and a helper
  - Make sure both take notes of their actions and observations

If a person sitting next to you suddenly fell ill with a heart attack, what do you do first? Hopefully, you answered that you would alert the emergency medical system! Even though being treated in the first minutes from when your heart stops is your best chance of recovery, the time spent asking for help is not wasted.

Your incident-handling team should have a senior member of management as its sponsor. This manager can help to clear out obstacles when you are under fire. To do that, you should strive to find a sympathetic senior manager, such as a Chief Information Security Officer (CISO), Chief Information Officer (CIO), senior Legal Counsel, or another related position that makes most sense in your organizational structure.

Always let this management sponsor know that you are in incident mode, either via e-mail or, for a more serious incident, with a phone call or visit. If you do not have a formal incident-team reporting structure, advise your manager and the security point of contact at a minimum. I cannot count the number of times that, at ten or so minutes into the incident, I realized I was over my head and needed reinforcements or specialists. It takes time to mobilize people; as soon as the incident is identified, you may wish to put them on alert.

One of the things that I am learning is that I can't do it all. If you are the primary handler on site, it is really a challenge to take notes, secure the area, and so forth. This is not a lone ranger sport. Realize you will probably need help and arrange for it in advance. Assign a minimum of two people to each incident: a primary handler and a helper. Have them both take notes independently of the other. Sure, there might be some conflicts in their notes. However, I'd rather take that small chance while avoiding crucial evidence slipping through the cracks!

© 2016 Ed Skoudis, John Strand

Notify Appropriate Officials and Create an Incident Tracking Entry

Containment

- Notify your local or organizational incident-handling team
- Notify your manager and security officer
- Remember vertical *and* horizontal reporting
  - Inform management (of course)
  - Inform impacted business unit
- Create entry in incident tracking system
  - CyberSponse is a commercial IR tracking system
  - There's the free RTIR Incident Response Tracking tool at http://www.bestpractical.com/rtir/
  - The Orion Live CD includes templates, tracking forms, and more at http://sourceforge.net/projects/orionlivecd/

There is a dynamic tension in many organizations between security and line management. If this is an issue in your organization, it can be a good idea to make sure both groups are kept in the loop throughout an incident. Users should be aware that when handlers are under fire, they may drop important information. In a large-scale attack, a handler might see a message, think that he will get right to it, and have 60 other things come up. For this reason, it is good practice to encourage users to demand a reply.

This is also a good reason why everything reported to a help desk or incident-handling center should be given a trouble ticket; it helps prevent loss of critical information. From a user's perspective, the best examples of horizontal reporting might include the system administrator, help desk, and the incident-reporting structure. Talk about good news, bad news! If he reports to all three (and many will), then three different trouble tickets get created, and there is a significant chance that actions will be taken that could damage the forensics evidence. On the other hand, additional eyes on the problem can be a good thing. The best solution is for the incident-handling reporting structure to communicate closely with the help desk and system admin shops.

Some commercial Security Information Management (SIM) and Security Event Management (SEM) products include the ability to assign an incident number around a group of events to track them together. Some go so far as to include a collaborative environment for incident handlers to conduct analysis and share conclusions across a distributed team. One such product is CyberSponse. It is a commercial grade IR tracking system that greatly simplifies the sharing and centralized collection of IR data across your team. It can be found at http://cybersponse.com.

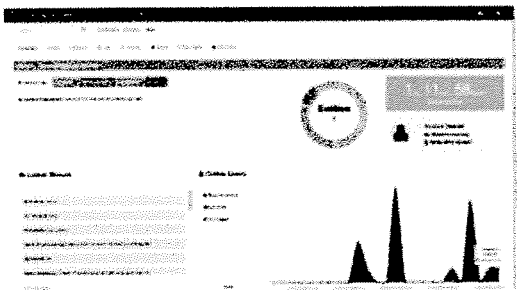
Alternatively, the Real Time Incident Response (RTIR) tool is a ticketing system targeted to incident-handling tracking. Its focus is on helping incident handlers stay organized when conducting their work, by providing an incident tracking number for each overall incident, plus tracking numbers for individual conversations.

Finally, the free Orion Live CD was specially designed for incident handlers and includes a variety of tools, templates, and reporting features geared to supporting incident response activities.

© 2016 Ed Skoudis, John Strand

CyberCPR — Containment

- Another option is CyberCPR
- Brainchild of fellow SANS Instructor Steve Armstrong
- Web app that tracks incidents, systems, and evidence
- Enforces need-to-know on incidents
- All files are hashed and encrypted upon upload



- Tracks user tasks and activity
- Tracks attacker campaigns
- Automates key analysis
- Secure real-time OOB chat
- Whilst a commercial tool, 1-3 users will be free

Another option is CyberCPR, the brainchild of fellow SANS Instructor Steve Armstrong and coded by several SANS Community Instructors and SEC504/560 and FOR408/508/610 alumni, it's designed around the DFIR user. Deployed as a hardened web application app (locally or with a cloud provider), it was designed to enforce need-to-know at the Incident level, allowing the existence of sensitive investigations to be supressed from those not involved.

It supports the incident handler by providing a central repository for evidence found and protects legal admissibility by encrypting and hashing (md5 and sha256) all data uploaded. This provides a secure alternative to e-mailing evidence and updates around the network. When system admins are provided accounts, they can upload large logs and files directly into the application, preventing what is known as evidence dispersal (where investigation evidence is scattered around various systems).

All items can have formal notes and comments added to them by that incident's users, and these notes are indexed and searchable. Incident tasks allows the Incident Manager to track actions they have delegated to other users. Together, these facilitate the secure coordination of evidence collection (the what), the recording of the reasons for that collection (the why), and required timeline of analysis (the by when). Once analysis is complete, logs and artefacts can be uploaded for others' review, notes, and comment upon the analyst's actions.

For out-of-band (OOB) communications, the tool includes both shout-box type chat and one-to-one messaging capabilities, allowing things like system admin or access passwords to be transmitted securely.

Best of all, as a strong community supporters, Steve and his team have committed that the 1-3 user version will always be free. This allows SMEs to benefit from the tool's commercially funded development and new features without breaking the bank (http://www.cyber-cpr.com).

© 2016 Ed Skoudis, John Strand

- Keep a low profile
  - Avoid looking for the intruder with obvious methods from the compromised machine (ping, traceroute, nslookup)
  - Don't tip your hand to the attacker
  - Maintain standard procedures
- Local handlers should keep making reports to the command center as they gather and analyze evidence

Rookie incident handlers can be spotted a mile away with a network logging system. They find an attack apparently coming from some IP address. So, they ping the address, then they do an nslookup and traceroute. Sometimes, they even telnet to it. You know, that might just tip off the attackers!

Some handlers feel that we should maintain normal procedures; if system backups normally took place at 02:45, then when 02:45 comes around, if at all possible, do what you usually do. This may matter in some cases of high value, but in general, the attacker couldn't care less.

© 2016 Ed Skoudis, John Strand

Short-Term (1) — Containment

- Try to prevent the attacker from causing more damage...
  - Without actually changing the compromised hard-drive data on the machine itself
- We want untainted evidence and can only get that through our image-creation process
- Some possible short-term containment actions
  - Disconnect network cable
  - Pull the power cable–loses volatile memory and may damage drive
  - Using network-management tools, isolate the switch port so the system cannot receive or send data... or place on an "infected VLAN"
  - Apply filters to routers and/or firewalls
  - Change a name in DNS to point to a different IP address
- You could also use WordWebBugs to track the attacker
  - These documents call back (preferably to a non-attributable system) so you can identify where your sensitive data is
  - They are built into the Active Defense Harbinger Distribution
  - http://sourceforge.net/projects/adhd/

For short-term containment, we just want to stop the attacker's progress, without making any changes to the impacted system itself. We want to keep the target machines' drive image intact until we can back it up. Therefore, this short-term containment typically involves disconnecting network access and/or power.

If you have the ability to control your switch infrastructure, you may want to consider isolating the switch port to which the impacted machine is connected, or even place that system on an isolated, infected VLAN so that you can still communicate with it, but it cannot infect other machines.

Another option is based on the fact that most attackers target systems based on their IP address (such as 198.167.22.13) and not their domain name (www.[yourdomainname].com). If this is the case, another option for short-term containment involves altering DNS so that the domain name(s) for the impacted system(s) points to a different IP address, perhaps one where you have a newly installed, secured machine offering up the desired production service. Once that new DNS address record has propagated, your users relying on the domain name will be accessing the new system at the new IP address. The attacker, if he or she is using an IP address to hit the target, will continue to go to the old IP address. You could place a honeypot at that address, simply null route all traffic going to the address, or just leave it as a completely non-responsive, unused address.

You could also use WordWebBugs to track the attacker. These documents call back (preferably to a non-attributable system) so you can identify where your sensitive data is.

http://sourceforge.net/projects/adhd/

© 2016 Ed Skoudis, John Strand

Short-Term (2) | Containment

- If short-term containment disables the system (such as removing it from the network and/or denying legitimate users access to the machine) ...
- ...make sure you advise someone in the business unit responsible for the system...
  - The information or application owner
- Advise them in writing, with a signed memo or at least an e-mail that gets acknowledged
- They may disagree with your advice to drop the system... in case you disagree, the business unit almost always wins!

Containment (both short- and long-term) might stop the system from performing various business actions. Therefore, make sure you get approval before taking action that will impact business. Call the business unit teams before dropping a system.

© 2016 Ed Skoudis, John Strand

ISP Coordination | Containment

- For external attacks, coordinate closely with your Internet service provider
  - It may be able to assist you in identification, containment, and recovery
    - Especially for large packet floods, bot-nets, worms, and virulent spam
  - Furthermore, the information you provide may save someone else a lot of pain
    - We need to work together as a community to foil widespread attacks
- Also, you may need to rely on someone else's ISP to get a bot-infected system taken offline

Most organizations realize they need to coordinate with their ISPs to bring some incidents under control, especially in large packet floods, wide-scale botnets, and their communications channel, worms, and virulent spam. Many ISPs are experienced at network-based exploit and denial of service types of incidents and are efficient at handling them. ISPs often keep system and even network logs, at least for a reasonable timeframe. Also, they may be able to spare other folks that get their service from the trouble that you had to go through.

By the way, you should be aware of one group that is extremely experienced in handling incidents; these are the computer and network staff of colleges and universities. As your organization is paying for employees' higher education and training, it can be a good idea to make contact with the computing staff at your local educational institution. If you build a relationship with them, they can really help you when you are under fire.

© 2016 Ed Skoudis, John Strand

Creating Forensics Images — Containment

- Make forensics images of affected system(s) as soon as is practical
  - This initial image will be used as a source for forensics analysis
- Grab an image of *memory* as well as the *file system*
  - Don't do graceful shutdown–you'll lose valuable data!
- The Volatility Framework by Volatility Systems and Memoryze by Mandiant include memory capture and analysis capabilities for Windows machines
  - We perform a hands-on lab of memory analysis later in class
- Use blank media
  - Old media often contain remnants
  - Newly purchased media may have some data on it, so beware
- If possible, make a bit-by-bit image to get all file system data
  - dd is your friend....
- Not all incidents will allow you to do a full backup and analysis
  - Time-sensitive incidents may require advanced network, domain, and live forensics
- Create a hash of the original and your images, such as an MD5 and SHA hash using md5sum, md5deep, or a forensics tool with hash-calculating capabilities

Failure to take complete notes is the most common error that incident handlers make.

Failure to make a good working forensics image is one of the most common errors. This is compounded because the incident handler is often called to work on a system that hasn't been backed up in a long time, sometimes years. Of course, the data is irreplaceable and mission essential. Many systems are being purchased today with multi-hundred gigabyte hard drives and no tape or other backup method. If you do not make a good forensics of the system before you start doing detailed analysis, you drastically reduce the chance of that system information being usable in court. The other attorney could claim that you modified the system. If you must do things on the system before backing it up, and sometimes this is necessary, try to log each command you type and the system's response.

This initial forensics image in the Containment phase serves as a source for forensics analysis. Make sure to get a copy of both memory and the file system.

You will not be able to make forensics images for all systems in all incidents. Some incidents may require you to rely on network domain and live forensics.

The ideal image, however, is the binary, bit-by-bit image; this gets everything on the disk, including deleted and fragmentary files. One of the most popular tools for creating binary images on UNIX and Windows machines is dd. I use it all the time, and it is a major component of the SANS Forensics Track (SANS Security 508) for gathering system images for analysis. It is built into many UNIX and Linux distributions, and is available in many incarnations for free for Windows. My favorite Windows version is available at http://www.gmgsystemsinc.com/fau/. Mandiant has released a powerful tool called "Memoryze" for capturing and analyzing memory on Windows machines. Volatility Systems released a tool called the Volatility Framework that likewise can be used to capture and analyze memory dumps.

© 2016 Ed Skoudis, John Strand

Drive Duplicator Hardware and Write Blockers

Containment

- Check if you have drive duplicator hardware
  - Make sure it copies bit-by-bit and not just allocated space
- Consider buying write-blocking hardware
  - Tableau (www.tableau.com) has several such solutions
- Destination drive should be bigger than source drive, typically by at least 10 percent
  - Otherwise, you may lose data due to difference in drive geometry

A hardware drive duplicator allows you to copy entire hard drives with ease. Drive duplicators are an essential tool for dealers, system builders, and incident handlers. The idea is to set up your SCSI or IDE drive with all your original software. If you use these devices to provision your system, there are no compatibility issues; this is how the computer was fielded in the first place. To copy, connect blank drives to the duplicator and press "copy." If you are not already using these to field new systems, take sample drives from your systems to make sure the duplicator meets your needs. Also, be certain this is a bit–bit copy mode; a good clue is if the duplicator is operating-system independent.

Alternatively, you may want to buy a write-blocking device, which lets you plug an IDE, SCSI, or other drive into the device as a source, and connect a firewire or USB drive to the device as a destination. By making the copy in a read-only fashion, your evidence is more trustworthy.

When copying to a new drive, make sure the destination drive is larger than the origination drive, to accommodate differences in the drive geometry. I like to make the destination drive at least 10 percent larger than the source drive.

© 2016 Ed Skoudis, John Strand

Determine the Risk of Continuing Operations

Containment

- Acquire logs and other sources of information. How far did the attacker get?
- Review logs from neighboring systems
- Make a recommendation for longer term containment
  - Document recommendation in signed memo
  - Ultimately, it's a business decision...
  - ...informed by incident handler's input

Make sure to watch out for trust relationships concerning the affected system. The most important trust relationship is often the desktop of the system administrator to the system and to other systems they administer. Look over the logs from nearby systems and trusted machines. Try to get a feel for how far the bad guy may have penetrated.

This is that critical moment: Do you down the box? Is this a contain and clean or a watch and learn? Most people contain at this point; however, sometimes the affected system(s) are crucial to your organization's operations. While you probably wouldn't want to keep running them at all costs, it makes sense to accept a higher risk in such circumstances.

Remember, the ultimate decision for downing the machine is a business call. Make a recommendation, which should be documented in a signed memo to the business owner of the machine. Realize that you won't always get your way on such decisions.

© 2016 Ed Skoudis, John Strand

Long-Term Containment

- Once we've got our back up for forensics analysis, we can start making changes to the system
- Therefore, we can implement longer-term containment strategies
- *Ideal:* If the system can be kept offline, move to the Eradication phase
  - Get rid of the attacker's stuff
- *Less-than-ideal, but-sometimes-necessary:* If the system must be kept in production, perform long-term containment actions

Long-term containment might change the drive image of the impacted system. But, that's OK, because we've already gotten our backup for forensics analysis.

Ideally, we can keep the compromised machine offline for an extended period of time, while we fully eradicate the attacker's artifacts, possibly by rebuilding the system or restoring it from a backup. In such cases, our long-term containment steps actually involve detailed eradication. In other words, after creating forensics images, we move onto the Eradication phase when extended downtime is acceptable. However, for critical production systems, extended downtime is usually not permissible. In such cases, the incident handlers must stay in the Containment phase, performing long-term containment actions on a live system to keep it running.

© 2016 Ed Skoudis, John Strand

Long-Term Actions

Containment

- Numerous potential actions, including
  - Patch the system
  - Patch neighboring systems
  - Insert Intrusion Prevention System (IPS) or in-line Snort
  - Null routing
  - Change passwords
  - Alter trust relationships
  - Apply firewall and router filter rules
  - Remove accounts used by attacker
  - Shutdown backdoor processes used by attacker
- Remember, you still need to do eradication...
- The idea for long-term containment is to apply a temporary band-aid to stay in production while you are building a clean system during eradication

There are several long-term containment activities, but the most likely, by far, is just patching the system if the attacker compromised it by exploiting a vulnerability. Handlers should also patch other nearby or similar systems to ensure that they do not get compromised. If patching is impractical over the short term, you may want to consider deploying an in-line Intrusion Prevention System, such as a commercial solution or even in-line Snort, which can block some attacker activity. Other long-term containment options that allow us to keep the system in production include null routing, in which we configure routers to drop packets associated with a given source or destination IP address used in the attack. You may need to change passwords for accounts to introduce a discontinuity in the attacker's access. Likewise, trust relationships between machines may need to be altered and/or broken so that an attacker with access to one environment cannot simply access another set of systems without re-authenticating. Firewall rules and router access control lists may need to be tightened to prevent deeper attacks as well. Of course, you may need to remove accounts created by the bad guy and kill any processes that offer the attacker backdoor access of the machine.

Don't think you're done with the incident-handling process, however, just because you applied a patch! You've still got the Eradication, Recovery, and Lessons Learned phases.

© 2016 Ed Skoudis, John Strand

Continue to Consult with System Owners — Containment

- Keep system owners and administrators briefed on progress
- Don't play the "blame game"
  - Never allow fault to be an issue during incident handling
  - Assigning fault now closes down important avenues of investigation
  - Sometimes, as you learn more, assumptions change
  - If fault absolutely must be assigned, do that during the Lessons Learned phase (Phase 6), not the Containment phase

One of the goals of a top-notch incident handler is to be as low impact as possible on the folks who own the computer with the incident. In some sense we are guests, or servants. We will be with the system for a short time, but it is not ours; we don't depend on it or administer it. It is important to keep the system administrator and the owner up-to-date on the situation. If you do not, they will try to get information from the command center and they may decide that they do not trust you. When this happens, it is harder for everyone.

Very often, assumptions change as more information becomes available during an incident. Early assumptions are often proved wrong. If you were to blame an individual and the facts later showed that person was not at fault, your credibility would be lost, at least in that part of the organization. There is more than just credibility at work here. Even though you are there to help, in some way, you are identified with your organization's security forces and you have access to their secrets. It is really important not to foster resentment; you need their trust and support to do your job.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. Incident-Handling Process
3. Preparation
4. Identification
   - Lab: Windows Cheat Sheet
5. Containment
6. **Eradication**
7. Recovery
8. Lessons Learned
   - Lab: Enterprise-Wide Identification and Analysis

Now, we turn our attention to what is probably the hardest problem in incident handling: Complete and safe total removal of any malicious code and other artifacts left by the attacker on the system, such as pirated software, pornography, and other illicit data. Although malicious code is not an issue in many incidents such as fires, or Internet delivered denial of service, this is one of the hardest problems a handler faces. This is why the GIAC Advanced Incident Handler Certification (GCIH) invests so much time covering malicious code.

© 2016 Ed Skoudis, John Strand

Eradication

- With the bleeding stopped, the goal of the Eradication phase is to get rid of the attacker's artifacts on the machine
- Determine cause and symptoms of the incident
  - Use information gathered during identification and containment
  - Try to isolate the attack and determine how it was executed

Now, with the bleeding stopped, the goal of the Eradication phase is to get rid of the attacker's artifacts on the machine, including accounts, malicious code, pirated software, porn, or anything else the bad guy left on the machine.

Reformatting and reinstalling the operating system from scratch may be considered a valuable shortcut in the handling process. Although it is certainly true that total destruction of the contents of the disk take care of any malevolent code, the opportunity for re-infection via the same channel after you reload the operating system still exists. There are many cases where handlers have taken systems down and reloaded the operating system only to have the box compromised again a few days later. The best course of action is to determine what the cause of the incident was, to find the vector of infection, and take action to prevent this from happening again.

The network system and forensic skills needed to do this are difficult to develop and, while we continue to try to make top-quality training available, there is no substitute for actual experience.

© 2016 Ed Skoudis, John Strand

Restoring from Backups — Eradication

- Locate the most recent clean backup
  - Search for a recent backup before an intrusion
  - In case of a RootKit-style attack (which modifies the operating system itself)...
  - ...wipe the drive (zeroing it out), reformat, and rebuild the system from the original install media and patches
    - Without a complete reformat, the attacker's residual data, tools, and access may linger

Backups simply do not happen as often as they should. They also are not tested in some cases. If an affected system has a recent clean backup, and you can identify when the compromise occurred, recovery is a matter of wiping the drive (zeroing it out), reformatting the drive, reinstalling the operating system, reloading the data from backup, adding any lost data and fixing the vulnerability that caused the problem in the first place.

If no backup is available, loading data by hand to a USB re-installed OS is a tricky and expensive process. If your organization has a central incident-handling team, you may want to establish a policy that this is the responsibility of the affected group or department.

In any case, here is the bottom line of eradication: There was an incident, you show up, you save the day, you are a hero! This is great. The problem comes back six hours later, and you are a goat! This is not great. Err on the side of caution; make sure all parties know if the safest course is to down the system and scrub it, they are making a risky choice. Don't keep doing business exactly like you were before being compromised.

© 2016 Ed Skoudis, John Strand

Removing Malicious Software — Eradication

- Remove malware inserted by the attacker
  - Virus infestations
  - Backdoors
  - RootKits and Kernel-Level RootKits
- If you have a RootKit or Kernel-Level RootKit, you should rebuild from scratch
  - Format the drive
  - Operating system (and PATCHES!)
  - Applications (and PATCHES!)
  - Data (the hardest one... possibly tainted back up)
- Encourage the impacted business unit to rebuild, reviewed by the computer-security team (including incident handlers)
- Unfortunately, there may be times where the attacker did not use malware
- Attackers often use services such as SSH and Remote Desktop

Viruses are fairly interesting. They are easy to deal with after the antivirus companies have analyzed them. You just let the software clean up the problem, and it does a great job. Dealing with viruses that don't yet have antivirus signatures is a much harder problem.

If the attackers change the operating system itself, by installing a RootKit, you should rebuild from the original install media, and install patches. Make sure you patch the system thoroughly, or the attacker will return.

Also, encourage the impacted business unit to do the rebuild, under your supervision. That way, you can make sure they understand the build process. Furthermore, they can verify that the system is functioning properly. Finally, you can verify that all patches are installed when they rebuild the machine.

It is also common for an attacker to use common and legitimate services, such as SSH and Remote Desktop, to persist and spread. It is incredibly important to monitor the logs of these services for irregularities, like strange source IP addresses and multiple concurrent logins from a single user ID.

© 2016 Ed Skoudis, John Strand

Improving Defenses | Eradication

- Implement appropriate protection techniques
  - Applying firewall and/or router filters
  - Moving the system to a new name/IP address
  - Null routing particular IP addresses
  - Changing DNS names
  - Applying patches and hardening the system

Once your system is hacked, the word gets out and every pea-brained hacker on the planet lines up to take another shot at you. It is not enough to simply recover the system; the security of the affected system(s) needs to be upgraded. If it is a production system, you may hear arguments that the organization cannot risk modifying a production system. This is a valid and important argument. The flipside is that, if the system was compromised, it must have some vulnerability. If we do not remove the vulnerability, the system may become compromised again.

The simple trick of changing the name and IP address of the system can solve many problems.

© 2016 Ed Skoudis, John Strand

Vulnerability Analysis | Eradication

- Perform vulnerability analysis
  - Perform system vulnerability analysis
  - Perform network vulnerability analysis
  - Search for related vulnerabilities
  - If possible, scan your entire network for interesting ports with a port scanner, such as Nmap
  - A vulnerability scanner, such as Tenable's Nessus, OpenVAS, Rapid7 NeXpose, and Qualys can also be a big help
- Remember that attackers often use the same exploit and backdoors on multiple machines
  - Look for them throughout your environment

Vulnerability scanners, such as Tenable's Nessus, OpenVAS, Rapid7 NeXpose, Qualys, and others, can identify weaknesses in your organization's internal network. Nessus and Nmap, two free tools, are among my favorites for scanning.

After placing a suspect system on a small hub and doing the backup, I have sometimes found it helpful to run Nmap on the target computer from another system on the hub. Several times, this has given me insight into the potential problems I may be dealing with by showing unexpected listening ports.

Running a security scanner on the neighboring systems in a compromise can help you make sure you have full and complete eradication. If one system is compromised, there is every chance the number is actually two or more.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. Incident-Handling Process
3. Preparation
4. Identification
   - Lab: Windows Cheat Sheet
5. Containment
6. Eradication
7. **Recovery**
8. Lessons Learned
   - Lab: Enterprise-Wide Identification and Analysis

Now, it's time to get back in business. That's what the Recovery phase is all about.

© 2016 Ed Skoudis, John Strand

Validation | Recovery

- The goal of the Recovery phase is to put the impacted systems back into production in a safe manner
- Validate the system
  - Once the system has been restored, verify that the operation was successful and the system is back to its normal condition
  - Always ask for test plans and baseline documentation
  - Run through the tests, or better yet, have the business unit retest

Remember that after you have touched the machine, everything that breaks is your fault. Be sure to get the owner of the machine to sign that it is back in full operation. Make every effort to ensure that the system is working properly before leaving the scene. If some functionality is not present, the default stance is usually to blame the incident-handling team in some organizations. You need to proactively avoid such a situation by having the business unit test the machine before going back into production.

© 2016 Ed Skoudis, John Strand

Restore Operations

Recovery

- Decide when to restore operations
  - Try for an off-hours timeslot
    - It's easier to monitor carefully
    - You will often be over-ruled on this, with the business opting to restore service immediately once systems are rebuilt and ready
  - Put the final decision in the hands of the system owners
  - Provide your advice, but they make the final call
    - Document your advice in a signed memo

The decision of when to put the system back into business has to be made by the system owner. As a handler, you can give them advice and be helpful, but this is their call. They are the ones that depend on this system.

Document your advice in a signed memo to the system owner. Remember, you may not get your way in this decision, but at least you document your recommendations.

© 2016 Ed Skoudis, John Strand

Monitor | Recovery

- Monitor the systems
  - Once the system is back online, continue to monitor for backdoors that escaped detection
  - Utilize network and host-based intrusion detection systems and Intrusion Prevention Systems
    - If possible, create a custom signature to trigger on the original attack vector because attacker will likely try same thing again
  - Also, carefully check operating system and application logs

Needless to say, if the eradication was not complete or the infection vector was not closed off, the earlier you detect re-infection, the better off everyone is. It is also politically better if the handlers detect the problem and show up to fix it than if the problem comes to light because business operations are affected. This is a serious problem. Many times, handlers take some shortcut along the way, or there is something you never discovered about the attack vector, and the problem comes back.

© 2016 Ed Skoudis, John Strand

Looking for Artifacts to Come Back — Recovery

- One of the most important things handlers can do during recovery and follow-up is to check regularly for re-compromise
- Note that attackers don't always use malware; sometimes, they log in via normal mechanisms for which we can look
- We urge you to write a script that checks whether the artifacts left by the attacker have returned, and run the script daily (or even more often) for several months
  - Look for changes to configuration via registry keys and values
  - Windows reg command
  - Look for unusual processes
  - Windows wmic or tasklist commands, or Linux ps command
  - Look for accounts used by the attacker
  - Windows wmic useraccount or net user commands, or Linux cat /etc/passwd
  - Look for other artifacts we discussed during the earlier lab
  - Look for simultaneous logins
- In other words, apply the cheat-sheet techniques, but now looking for *specific* items left by the bad guy to determine if he or she has *come back*
  - Over 100 examples of checker commands at www.commandlinekungfu.com

One of the most important things for incident handlers to do in the Recovery phase and in the months following an incident is to check regularly to see if the attacker has returned. Most human attackers that compromise machines do return to the scene of the crime, making the same or similar changes to the system that they made upon initial compromise. Some attackers do not use malware, preventing handlers from using malware detection to identify the attacker's return. Instead, these attackers use normal login mechanisms to access the system and reconfigure the machine to maintain control. We need to look for these changes aggressively to detect an attacker's return.

Because of this fact, we urge handlers to write up one or more simple scripts that they can run on a daily basis to look for artifacts left by the attacker in the previous compromise cycle. If the attacker comes back and recreates the same artifacts, your script should detect that fact and notify you. We recommend running the script daily or even more often.

Your script could look for changes to the configuration of a Windows machine by checking for changes to registry keys and values using the reg command, which works remotely using the "reg \\[MachineName]" syntax. For example, if the attacker alters the configuration of an application via a registry key so that it logs credit-card numbers, make sure you look for that kind of change in the registry.

Or, you could look for unusual processes using the wmic command (which works remotely when run with "wmic /node:[MachineName] /user:[Admin] /password:[password]") or the tasklist command (which can be run remotely using the psexec command from Microsoft Sysinternals). On Linux, you could us the ps command to get a list of processes.

Also, look for accounts the attacker created, which can be pulled via wmic with "wmic useraccount list brief" or the "net user" commands. On Linux, you can get this information via "cat /etc/passwd."

Likewise, you should look for other artifacts covered by the Intrusion Discovery cheat sheets. The good news is that your script can look for *specific* items created by the bad guy during the earlier incident, so you can focus in on detecting certain changes to identify the attacker's return. The blog at www.commandlinekungfu.com includes more than 100 examples of commands that check various aspects of system status for Windows and Linux.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. Incident-Handling Process
3. Preparation
4. Identification
   - Lab: Windows Cheat Sheet
5. Containment
6. Eradication
7. Recovery
8. **Lessons Learned**
   - Lab: Enterprise-Wide Identification and Analysis

Suppose an incident was detected at 4:30 PM on a Friday afternoon, just as you were hoping to slide out the door. You stayed all night and finally got control of it at about midnight. While it was going on, you had all that adrenalin; you felt like you were part of something pretty darn exciting and now you just wish it would go away! The adrenalin has worn off, you are tired; in fact you are sick of the whole situation and would prefer to live your life without ever hearing about it again!

It takes discipline to do the report. It takes even more to attend a lessons learned meeting, but it is important. Your attackers are improving. We have to improve as well. One way to improve is to learn from our mistakes and move on to make new mistakes instead of repeating the old ones. This is the primary purpose of the follow-up part of the process!

© 2016 Ed Skoudis, John Strand

Report | Lessons Learned

- The goal of the Lessons Learned phase is to document what happened and improve our capabilities
- Develop a follow-up report
  - Start as soon as possible... right after recovery (i.e., going back into production)
  - Assign the task to the on-site team
    - Supervised by the head incident handler
  - Include forms from www.sans.org/score/incidentforms
  - Encourage all affected parties to review the draft
  - Attempt to reach consensus and get sign off
  - In the unlikely event that someone doesn't agree, have him submit and sign off on his own version of the events

The only one that really can or will write the report is the on-site handler. The handler submits the draft to the head of the incident-handling team. This chief edits the document and interacts with the handler to make sure the document reflects what actually occurred, in light of the organizations' culture. We should allow everyone involved to review the draft. Have everyone involved in handling the incident sign off on the report, agreeing to its contents.

If anyone has a strong disagreement about the facts of the matter, he can submit that, and his statement can remain a part of the incident record. It is far better to find out that you have a lack of consensus before going to court than during court!

© 2016 Ed Skoudis, John Strand

Meeting — Lessons Learned

- As soon as practical (within two weeks of resuming production)
- Review the report
- Finalize Executive Summary
- Keep it short and professional
  - Maximum length: Half day

After the report has been reviewed, schedule a Lessons Learned meeting. In general, the main purpose of such a meeting is to get consensus on the Executive Summary of the report. This meeting should occur within two weeks of resuming production, while the events and report are still fresh in people's minds.

What is the most important thing for an Executive Summary to cover? How much the organization saved by having an effective incident-handling procedure and team!

During every incident, mistakes occur. We learn from these, improve our process for the future, and move on. Sometimes, we run into policy or other organizational problems that hinder bringing the incident to a close. We note these and submit them to management for its consideration.

Follow-up meetings are never the most popular of events. Everyone is tired; they have been under stress. The system is now back in operation and the last thing anyone wants to do is have a meeting to rehash painful memories.

However, this is a valuable tool for organizational improvement. This is the hardest time not to blame people. The focus should be on process improvement.

© 2016 Ed Skoudis, John Strand

Apply Fixes | Lessons Learned

- Based on what you learned, get appropriate approval and funding to fix
  - Your processes
  - Your technology
  - Improved incident-handling capabilities

We want to have constant improvement, so the cause of the incident can be eliminated or minimized. You must go to management and make a compelling case for fixing the problem that caused the incident in the first place. This may mean an alteration to the processes or technology in your environment.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. Incident-Handling Process
3. Preparation
4. Identification
   - Lab: Windows Cheat Sheet
5. Containment
6. Eradication
7. Recovery
8. Lessons Learned
9. **Enterprise-Wide IR**
   - Lab: Enterprise-Wide Identification and Analysis

Suppose an incident was detected at 4:30 PM on a Friday afternoon, just as you were hoping to slide out the door. You stayed all night and finally got control of it at about midnight. While it was going on, you had all that adrenalin; you felt like you were part of something exciting, and now you just wish it would go away! The adrenalin has worn off, and you are tired; in fact, you are sick of the entire situation and would prefer to live your life without ever hearing about it again!

It takes discipline to do the report. It takes even more to attend a Lessons Learned meeting, but it is important. Your attackers are improving. We also have to improve. One way to improve is to learn from our mistakes and move on to make new mistakes instead of repeating the old ones. This is the primary purpose of the follow-up part of the process!

© 2016 Ed Skoudis, John Strand

Enterprise Incident Response | Enterprise-Wide IR

- Determining if one system is compromised can be difficult
- Doing this at scale across thousands of systems can seem impossible
- With the right tools and techniques it is possible
- Many of the data points are already being collected in your environment
- You just need to know where to look

There are many different tools at the disposal of any incident responder who is working a large-scale incident. Some of these tools are commercial and others most likely are being used actively by your enterprise right now.

However, the goal of any enterprise-wide incident response engagement is the same: to identify indicators of compromise across multiple machines and user accounts.

The vast majority of environments are already collecting logs and data for many of the components we will discuss over the next few slides. It is just an issue of tapping into those data sources and extracting the proper data.

© 2016 Ed Skoudis, John Strand

Ingress and Egress Data

Enterprise-Wide IR

- Need to start somewhere
- Connection data from the various points of presence is a good start

1. **Web Proxy**
2. **DNS Cache**
3. **Connection Logs**

When we are working any incident, the first thing our responders ask for is the logs for the egress connections from the firewall, the DNS cache or DNS logs and, finally, the logs from whatever external web-filtering device the organization is using.

There are a couple of reasons for this. First, the logs from these devices work well as they serve as filter points for all traffic leaving the environment. Remember, in situations like an incident, the goal is not to stop an attack from coming in, but rather to be able to detect command and control points and identify additional internal systems that may be compromised. And, if a network is configured properly, all egress traffic should either flow through an egress firewall, resolve a domain via DNS, or connect through a egress web proxy.

© 2016 Ed Skoudis, John Strand

DNS Data

Enterprise-Wide IR

- DNS can be very powerful
- Simply reviewing a DNS servers logs and cache can reveal systems that are connected to known bad IP addresses and domains
- dns-blacklists.py can review DNS logs or cache to look for known bad IP addresses and domains
- Malware Domain List is a great site for updated lists of known bad actors

DNS data can be one of the most powerful tools you have for detecting malicious traffic leaving your environment. First, there are many examples where traditional AV fails to detect well-known malicious programs from running. Second, many of the domains used by well-known botnets, and C2 channels tend to be more static than the code base for the malware. Finally, it is relatively easy to compare the logs and/or the current cache for your DNS server with well-know evil domains and IP addresses.

One of the tactics we use as part of initial incident response is to compare the current cache of the DNS server with a list of evil IPs and domains by using a tool like dns-blacklists.py. One of our favorite lists to compare with is the one from Malware Domain List.

You can find Ethan's tool here:

https://bitbucket.org/ethanr/dns-blacklists

And Malware Domain List here:

http://www.malwaredomainlist.com/mdl.php

© 2016 Ed Skoudis, John Strand

Web Proxy Data

Enterprise-Wide IR

- Many organizations do not review these logs
  - Often, this is due to HR issues
- However, regular review can uncover compromised systems that are connecting to known bad C2 sites
- Review the length of URLs being visited
  - Many malicious URLS are very long
- Review user agent strings
  - Many malware specimens use older or odd user agent strings

Most every enterprise environment today has some sort of web-proxy content filter to restrict employee access to sites with objectionable content. Further, these can be powerful incident response tools for any responder.

There are a couple of things to look for. First, you can once again see if there are any IP addresses or domains being accessed that are known bad actors. Sure, many of these providers have their own regular updated black lists. However, it is good to get a second, or even a third, opinion.

Next, look at the length of the URLs being accessed. Many variants of malware will use long encoded URLs either as a command and control mechanism or as a way to deliver payloads. However, be careful because many legitimate sites also use long URLs. We recommend using this in conjunction with other indicators of compromise. For example, let's say one of your systems pops an AV alert. You then notice that some of the domains it is accessing are odd. Then, look at the URLs as another possible investigation item. This also has the added bonus of potentially identifying additional systems that are also compromised in the process.

Finally, review the user agent strings. If most of your systems are Windows 7 with the newest version of Internet Explorer, and you start seeing Windows XP with IE version 6, this could be malware. At the least, it is a system that needs to be upgraded. The reason for this is that some malware uses old user agent strings to "blend in." However, over time, they forget to upgrade or modify them.

© 2016 Ed Skoudis, John Strand

Connection Data

Enterprise-Wide IR

- Sifting through thousands of packets can be daunting
- However, reviewing Netflow data can reveal interesting patterns in connection statistics
  - Systems beaconing out every 30 seconds
  - Systems beaconing out at random intervals
  - Connections which live for far longer than they should
- Excellent tool by Eric Conrad for looking for persistent beaconing connections
- Hunting for evil actors is a big part of SANS 511: Continuous Monitoring

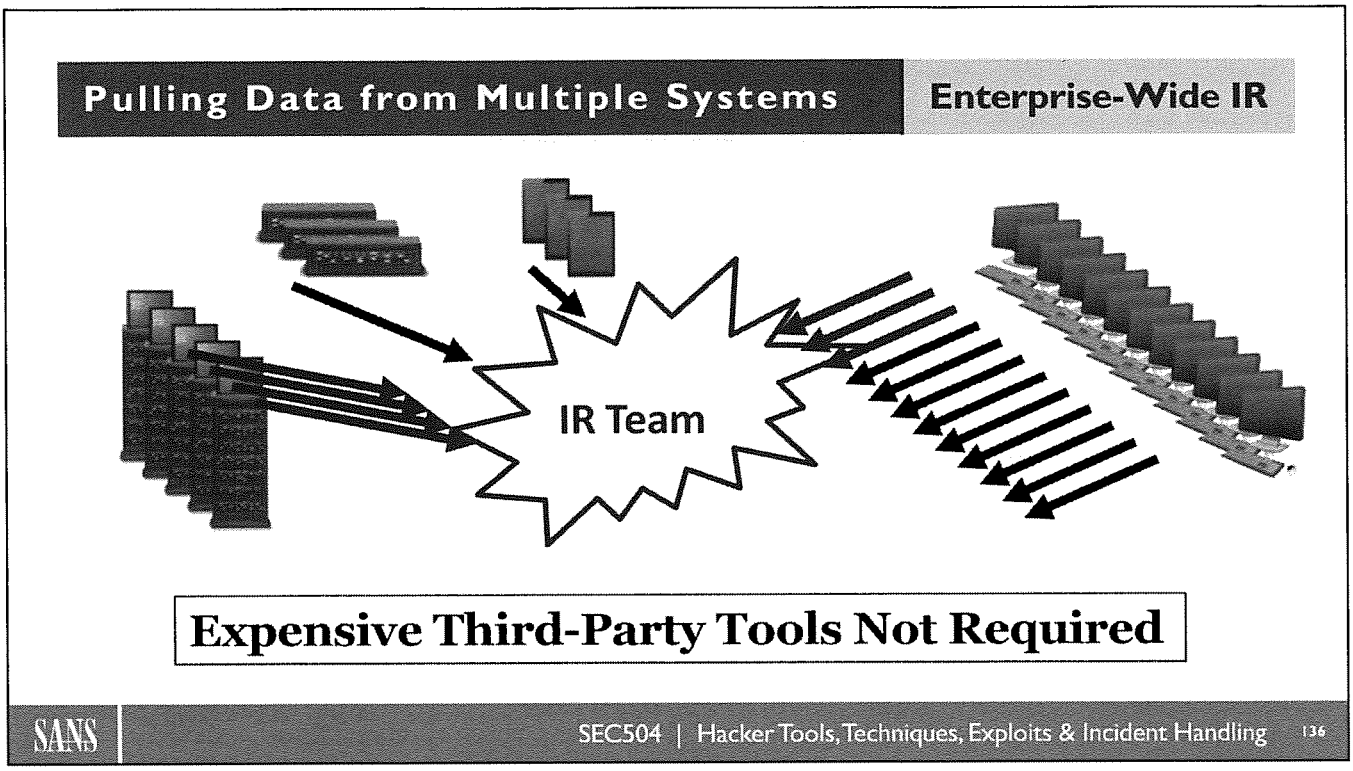
Another, often overlooked, component is seeking out beaconing data. It is often assumed that malware makes a persistent connection back to the bad guys' command and control systems. However, with today's modern malware, this is just not the case. Rather, malware today tends to connect back at regular (or irregular) intervals. In order to detect this connection method, it requires the analysis of the connection logs of your egress firewall that performs Network Address Translation (NAT) from internal, RFC 1918 IP addresses to externally rotatable IP addresses. Eric Conrad has an excellent tool to parse these log files to find connection attempts.

It can be found here:

http://tinyurl.com/505and511

It is also part of the SEC 511 Continuous Monitoring class.

© 2016 Ed Skoudis, John Strand



Next, we need to have a way to extract data from multiple internal computers to look for indications of compromise.

There are a number of useful techniques, and not all of them require an investment of third-party software. In-fact, many of these tools are already incorporated in most enterprise environments.

© 2016 Ed Skoudis, John Strand

WMIC

Enterprise-Wide IR

- Let's pull everything
  - C:\> wmic product get name,version
  - C:\>wmic /node:@systems.txt product get description,name,vendor /format:csv > SoftwareInventory.txt
- The /node:@systems.txt allows you to run the same command on multiple systems

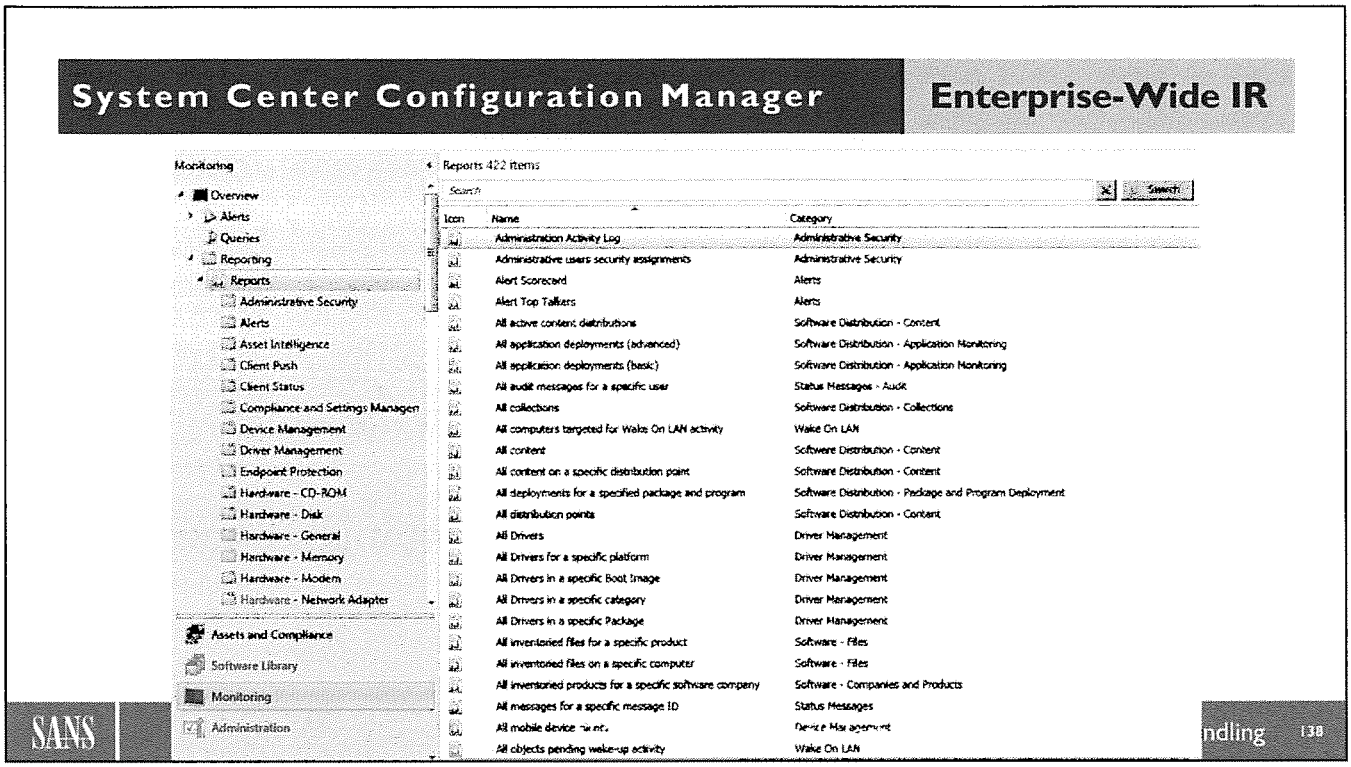
All the commands we ran in our Windows command-line lab can be run against multiple system

Earlier, we discussed the awesome power of WMIC. There is no greater tool for quick fire incident response than WMIC. During the Lunch lab, you were able to see that WMIC can query just about everything. However, the real power of WMIC is /node switch. With this switch, you can run WMIC commands on multiple systems at once and have the data reported back to you in CVS or XML format for easy parsing.

It requires you to be a domain admin in order to be completely effective. This is a consistent theme for the all of these tools.

Above are just some of our favorite IR WMIC commands.

© 2016 Ed Skoudis, John Strand



Just to be completely clear, SCCM is not easy to install. However, once it is up and running, it can be the most powerful weapon in your IR arsenal.

The most powerful IR component of SCCM is its reporting. It has the ability to inform you about the installed software on a system. It can pull drivers. It can pull users and services. It is literally a one-stop shop for enterprise IR.

Here is an excellent link for step-by-step instructions on how to configure SCCM and its reporting components:

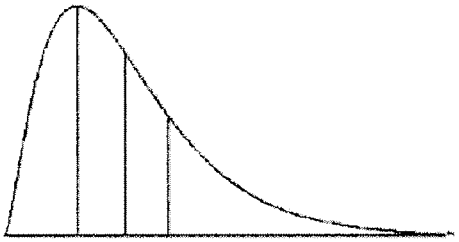
http://www.windows-noob.com/forums/index.php?/topic/4550-using-sccm-2012-rc-in-a-lab-part-11-adding-the-reporting-services-point-role/

© 2016 Ed Skoudis, John Strand

What About PowerShell?

Enterprise-Wide IR

- There is a wide variety of excellent PowerShell tools for incident response
- Let's focus on Kansa, an excellent detection tool from Dave Hull
- Kansa focuses on stacking like systems against each other to provide a ranked listing of processes, network connections, and configurations of systems
- This is all part of statistical long-tail analysis



Mode < Median < Mean

Although there are a number of excellent built-in tools for incident detection and response, there are also a solid number of tools available that are written in PowerShell.

One of our favorites is a tool called Kansa by Dave Hull. It can be found here:

https://github.com/davehull/Kansa

This tool creates a stacked analysis of the installed software on your environment. Then, you focus on software and processes, which are installed only on a few systems. This is called "long-tail analysis" or "positive skew analysis" in statistics.

© 2016 Ed Skoudis, John Strand

Kansa Setup

Enterprise-Wide IR

- Kansa requires PowerShell 3.0
  - A simple domain-wide install of 3.0 is required
- For full functionality, install Handle.exe and autorunssc.exe from Sysinternals from Microsoft
- For the machine launching the scripts, install Logparser from Microsoft
- On all targeted systems, run the following command to enable Windows Remote Management:
  - C:\>winrm quickconfig
- Add all hosts you want checked into a text file and loaded in the Kansa-Master directory

Before we get started, some setup work is required. The required steps in the slide gets Kansa configured properly.

The major thing to take from this is that all the required components for Kansa to run properly are from Microsoft. It is fairly easy to pull these files down and disperse them via Group Policy to the systems in your environment.

Note that autorunssc.exe and handle.exe need to be installed to the C:\Windows directory so they can be found by Kansa when it runs.

Also, this type of baselining is critical for effective incident response. The point is, run this before an incident occurs. This is a powerful tool, but it is only as good as the admins who run it.

© 2016 Ed Skoudis, John Strand

Kansa in Action

Enterprise-Wide IR

- Kansa supports the ability to pull the total count for specific things, such as Auto Start Entry Points (ASEP)
- In the following example, notice the count column on the left side:

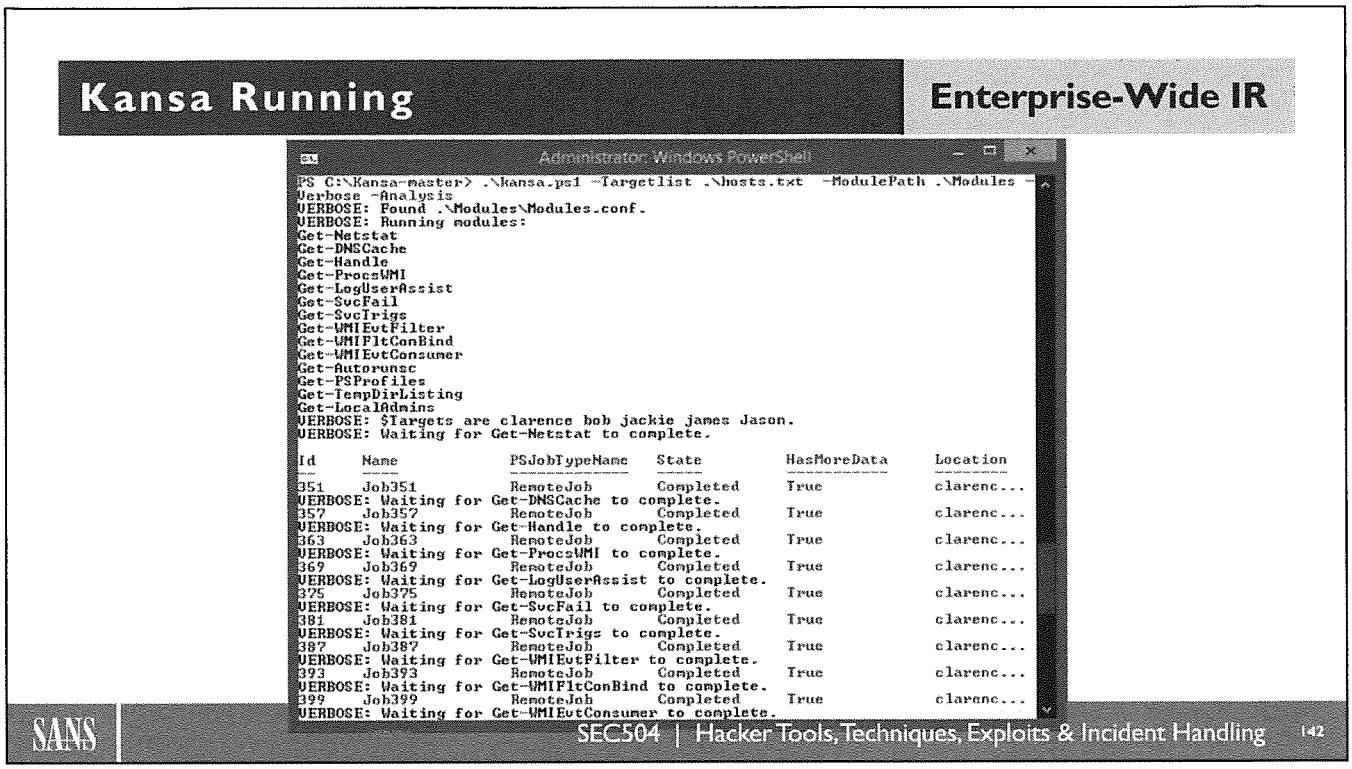
```
cnt Image Path                                                  MD5
--- ----------------------------------------------------------- --------------------------
10  c:\windows\system32\cpqnimgt\cpqnimgt.exe                   78af816051e512844aa98f23fa
10  c:\hp\hpsmh\data\cgi-bin\vcagent\vcagent.exe                54879ccbd9bd262f20b58f79c1
10  c:\windows\system32\cpqmgmt\cqmgstor\cqmgstor.exe           60668a25cfa2f1882bee8cf2ec
10  c:\program files\hpwbem\storage\service\hpwmistor.exe       202274cb14edaee27862c6ebce
10  c:\hp\hpsmh\bin\smhstart.exe                                5c74c7c4dc9f78255cae78cd9b
10  c:\msnipak\win2012sp0\asr\configureasr.vbs                  197a28adb0b404fed01e9b675e
10  c:\program files\hp\cissesrv\cissesrv.exe                   bf68a382c43a5721eef03ff451
```


In the slide image, there are a number of different auto-start programs pulled from ten different systems. Rather than attempting to review each system's ASEPs, you can simply look at the total count of ASEPs, which exist on all systems.

Ideally, you should be looking for one-offs, or entries that do not appear on all systems.

This can be a total nightmare in environments that do not practice consistent build processes and solid change management.

© 2016 Ed Skoudis, John Strand

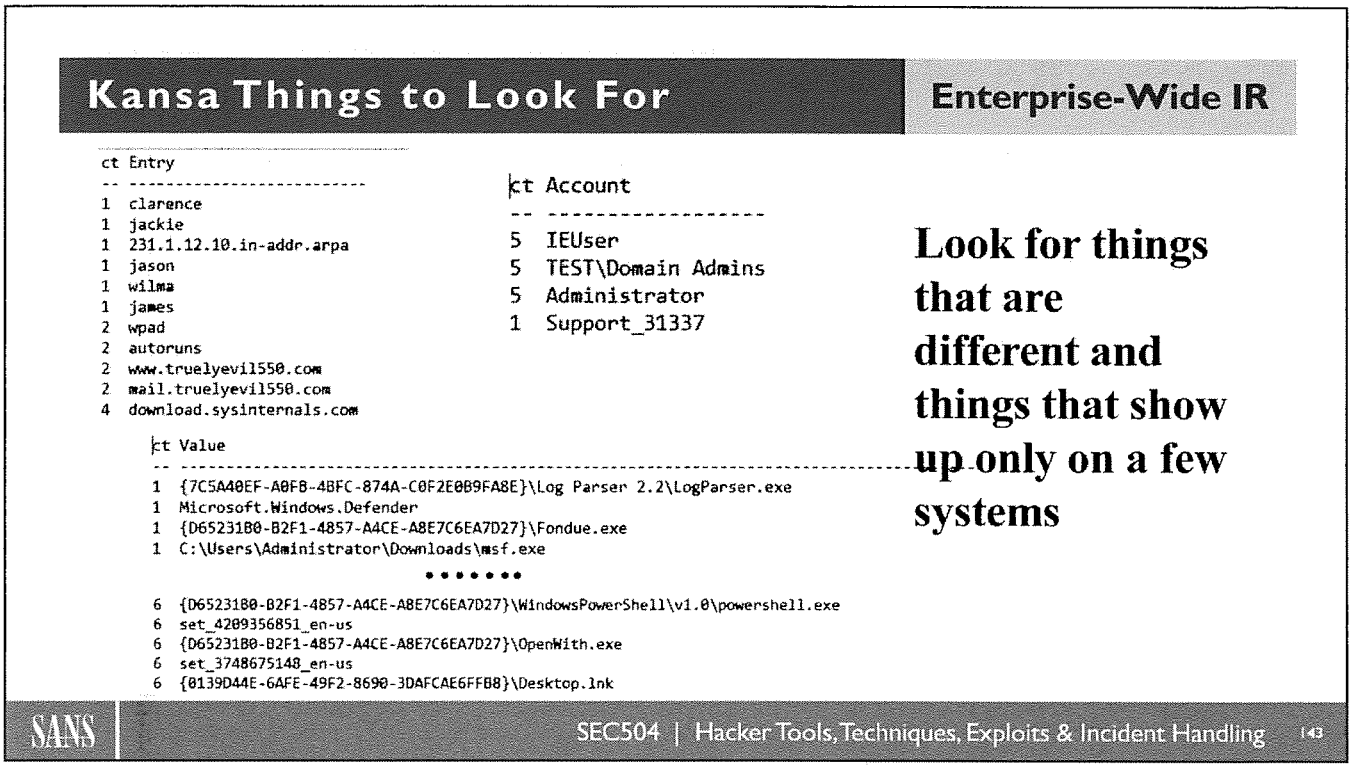


The slide image is what Kansa looks like when it is running properly.

Note that it can be run on multiple systems by using the –Targetlist switch with a file listing all of the systems you want it to run on.

Also, it is helpful to use the –Analysis switch. This triggers all the stacking components and analysis components. It is far easier to do incident response when comparing and contrasting systems against each other rather than looking at each system as a one-off.

© 2016 Ed Skoudis, John Strand



Ideally, malware jumps out at you. However, in reality, you are hoping to get two things from Kansa. First, you want to make the size of the data you are sifting through smaller. Think of the needle-in-the-haystack analogy. You are trying to make the haystack much smaller.

Second, you are trying to look at what exists on a smaller scale, that is, to look at what is installed and running on a few systems. This is where you will find malware and backdoors interesting.

Of course, your entire environment can be compromised with the same malware. Although this does happen, we usually find that bad guys use malware for initial compromise, and then use existing credentials to pivot and pillage.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

INCIDENT HANDLING

1. Incident Handling Definitions and Overview
2. Incident-Handling Process
3. Preparation
4. Identification
   - Lab: Windows Cheat Sheet
5. Containment
6. Eradication
7. Recovery
8. Lessons Learned
9. Enterprise-Wide IR
   - **Lab: Enterprise-Wide Identification and Analysis**

Now, let's have as lab about the major topics we covered so far.

© 2016 Ed Skoudis, John Strand

Lab: Enterprise-Wide Identification and Analysis

- Friday at 4:40 PM, you receive a call from a customer saying he is receiving spam from your organization
  - Something about selling timeshares
  - Attack happened on Thursday... He is telling you now
- The path in the SMTP headers checks out
  - The e-mails appear to be coming from your organization
  - This can be spoofed, but it still needs to be investigated
- It is up to you to properly identify this possible incident and determine what the overall risk to the organization is
- Time is of the essence, because attackers often reuse the same attacks against multiple target systems in an organization
- There is also a good possibility other systems may be compromised... Did we mention it is now 4:43 PM Friday?

Friday afternoon, your incident-handling team receives a call from a customer explaining that he is receiving spam from one of your users. He says it started at roughly 4:40 PM and involved timeshares in the State of Connecticut. The SMTP header checks out, and it appears the e-mail is coming from your network.

It is up to you to confirm the spam attack, check and see if the system in question is compromised, deduce how it was compromised, and discover if any other systems in your organization are also vulnerable to the same type of attack. Time is of the essence, because many times, attackers reuse the same attack on an organization.

© 2016 Ed Skoudis, John Strand

Lab: Roadmap

- Over the next few slides, we cover how to:
  1. Install Mandiant Redline
  2. Copy Windows\504_friday_ex\Friday.mans from the class USB to your C:\>tools directory on your Windows Computer
  3. Review the lab questions
  4. Review the evidence files in the Windows\504_friday_ex directory on the class USB
  5. Answer the lab questions
- Should you have any questions, ask your instructor for help
- Work together in teams of three people

Here are the overall steps to this lab.

You will review three different files in this lab. Two of the files are simple text files. One of the files is a .mans file. This is a memory analysis file created by Mandient's Redline tool used to analyze systems. You install this tool and review the data contained in the .mans file for indications of a compromised system.

Do not do these steps now. It is merely an overview of the lab process.

© 2016 Ed Skoudis, John Strand

Lab Setup: Installing Redline

- Mandiant Redline is an outstanding tool for memory analysis
  - It is on the Class USB in the Windows Directory: Redline-1.11.msi
  - If you need dotNetFx40_Full_x86_x64.exe, it is in the Windows Directory on the Class USB
- Simply navigate to the Windows directory on the Class USB and double-click the Redline-1.11.msi file
  - Follow the prompts to install Redline on your computer
- We use Redline to review a Redline memory analysis file
- The Redline analysis file is called Friday.mans and is in the USB Windows\504_friday_ex folder
- Copy the Friday.mans file from the Windows\504_friday_ex\Friday.mans file from the Class USB to your C:\>tools directory on your Windows computer
- If the C:\>tools directory does not exist, create it

In order to properly complete this lab, you first need to install Mandiant Redline. Redline is an outstanding memory-analysis tool. It is great for a quick first look at a system that has been compromised.

Because we do not have time for a full Redline analysis on a memory dump (it takes about 1/2 an hour), we have already done the initial Redline parsing and analysis of a memory dump. You need to install Redline, and then navigate to the Windows\504_friday_ex folder on the Class USB and open the Friday.mans file with Redline. This is the full Redline report on the memory dump from this attack.

Mandiant Redline is an essential tool for quick memory analysis. It reviews the various process and process calls to various Dynamic Link Libraries (DLLs) to see if there is any indication any of the processes are malicious. It is not a black list tool, but a tool that looks at memory to see if any processes are following common patterns used by many malware samples today. Sometimes, it works well. Other times, it may miss some malicious activity. We will be using it as a "quick analysis" tool, then in 504.5, we dig deeper into this memory dump with different tool called Rekall.

Simply copy the Friday.mans file from the Class USB to your C:\>tools directory on your Windows system.

To install Redline, navigate to the Windows directory on the Class USB and double-click Redline-1.11.msi. Then, simply follow the install directions.

If your Windows machine tells you that you need Dot Net 4.0, install it from the Windows directory on the Class USB.

© 2016 Ed Skoudis, John Strand

Lab Setup: Evidence

- All the evidence for this case is in the Windows\504_friday_ex folder on your Class USB
- Friday_Exchange.log
  - This is a log dump from your SMTP server
  - Check it for evidence of spamming from your environment
- Friday.mans
  - A memory analysis of a potentially compromised system
  - Copy this to your C:\>tools directory and double-click it *after* you install Redline
- Friday_SoftwareInventory.txt
  - This is an inventory of the software on systems that are part of the same LAN as the potentially compromised box
- Later, we provide steps to create these evidence files

All the evidence for this incident is in the Windows\504_friday_ex folder on your Class USB.

Friday_Exchange.log

The first file is a log dump from your Exchange server. This helps you confirm if one of your internal systems is compromised.

Friday.mans

This file is a memory analysis of the system believed to be compromised. You can open this file by simply double-clicking it ***after*** you install Redline, on your Windows system. **Once it opens, we recommend you look at the Timeline section under Analysis Data on the left-hand pane.**

Friday_SoftwareInventory.txt

This is a software inventory of all the systems on the same LAN segment as the alleged compromised system.

© 2016 Ed Skoudis, John Strand

Lab: Questions for Your Team

- Review the evidence files and answer the following questions:
  - Have we been compromised?
  - Which system has been compromised?
  - How was it most likely compromised?
  - Who is the user on that system?
  - Are any other systems at risk?
  - What are our containment next steps?
- Hints
  - Excel or another spreadsheet helps, but is not necessary
  - You do not have to know every field of every log file
  - Look for patterns
  - Work as a team... Sometimes talking through what you see helps
- Full walkthrough of all the evidence is on the following slides

Here are a few questions that need to be answered as soon as possible. Feel free to fill your answers on this page as you analyze the evidence and follow the steps of the lab.

1. Have we been compromised?

2. Which system has been compromised?

3. How was it most likely compromised?

4. Who is the user on that system?

5. Are any other systems at risk?

6. What are our containment next steps?

Just a few hints before we get started. First, if you have Microsoft Excel installed, it can be helpful for sorting through the data provided. However, it is not necessary. This entire lab can be done with Redline and a text editor, like Notepad.

Second, you do not have to know every line and every field in a log file. Simply scroll through the data and see if anything looks strange. Don't panic if a format is strange.

Also, we have provided all the answers and a walkthrough in this course book. If you are new to this field, you can simply follow along with our guided walkthrough of the evidence to see the approach and answers to these questions.

© 2016 Ed Skoudis, John Strand

Lab: Friday_Exchange.log

```
1/9/14,6:28:40,192.168.49.144 ... -,+p01c12m075.lilliputmail.tgt,250,0,318,27,0,SMTP
1/9/14,6:28:40,192.168.49.144 ... +FROM:<robert@target.tgt>,250,0,53,40,0,SMTP,-,-,-
1/9/14,6:28:40,192.168.49.144 ... +TO:<ceo@target.ae>,250,0,24,21,0,SMTP,-,-,-,-
1/9/14,6:28:40,192.168.49.144 ... +TO:<president@target.ae>,250,0,30,27,0,SMTP,-,-,-
1/9/14,6:28:40,192.168.49.144 ... +TO:<coo@target.ae>,250,0,24,21,0,SMTP,-,-,-,-
1/9/14,6:28:40,192.168.49.144 ... +TO:<hr@target.ae>,250,0,23,20,0,SMTP,-,-,-,-
1/9/14,6:28:40,192.168.49.144 ... +TO:<aaron.miller@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<adam.schwab@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<alan.fickbohm@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<alan.todd@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<alan.bakeberg@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<alice.whitebird@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<alison.kiesz@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<amanda.schmitgen@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<amber.anders@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<amy.gorham@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<amy.iversen-pollreisz@state.sd.>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<amy.kainz@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<andrea.meyers@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<andrea.hewitt@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<andrew.fergel@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<andy.mobley@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<anissa.grambihler@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<anjar.voorhees@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<ann.larsen@externalemail.com>,,,,,,,,,,
1/9/14,6:28:40,192.168.49.144 ... +TO:<ann.hirsch@externalemail.com>,,,,,,,,,,
```

Client IP Address

Lots of externalemail.com Addresses

SANS | SEC504 | Ha... dling 150

When you first open the Friday_Exchange.log in a text editor, such as Notepad, you notice a header on the first line. This row describes what each of the columns mean. For a full explanation of what each header is, reference the following Microsoft site:

http://msdn.microsoft.com/en-us/library/windows/desktop/aa814385(v=vs.85).aspx

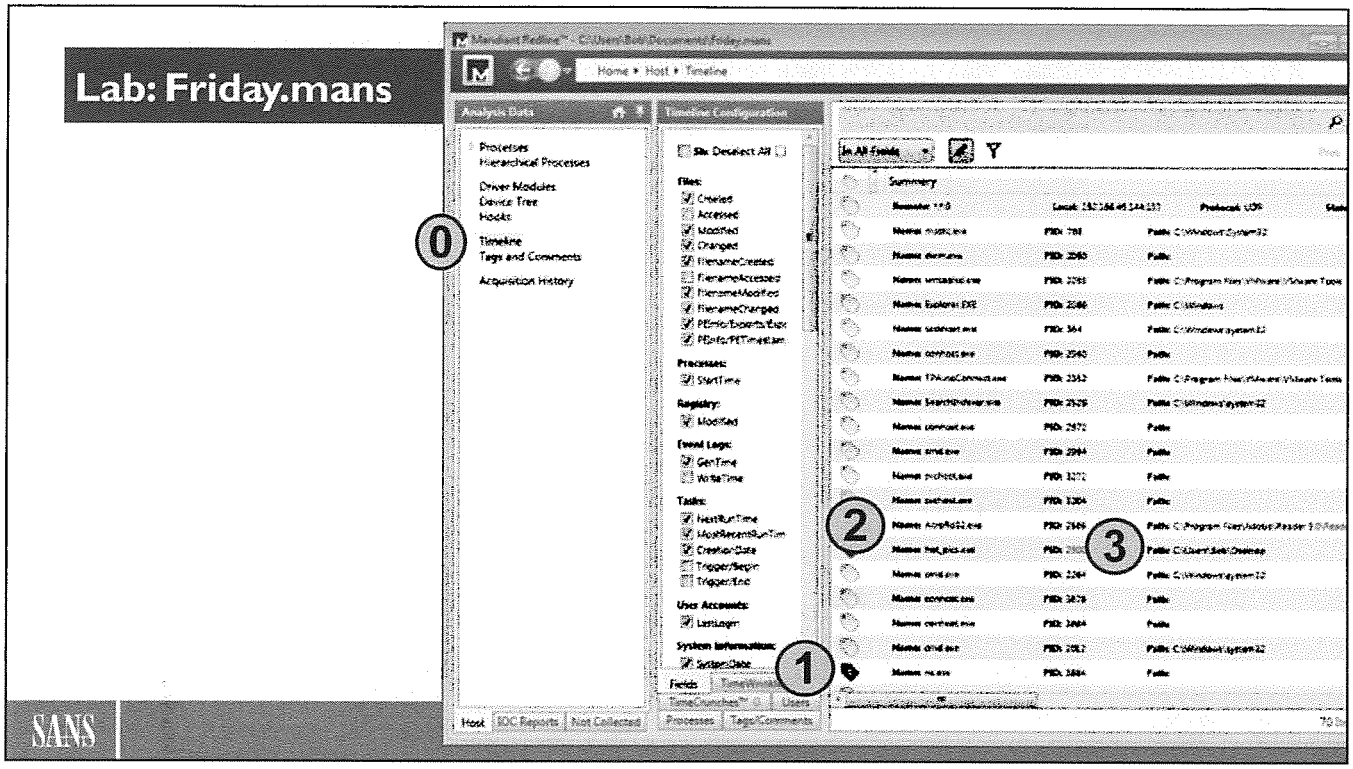
As said earlier, it is easy to get confused when first confronted with a log like this. Don't panic. If you look through the file, you see a large number of entries at the bottom called OutboundConnectionCommand with a large number of RCPTs. Also, if you look at the RCPTs, a very large number of them are not being sent to the target.tgt domain. Rather, they are being sent to a large number of externalemail.com addresses.

Now, look at the Client IP field for these entries (it is the third column in). Notice that all of these emails come from 192.168.49.144. This is a fairly good indication that this system is compromised and bouncing spam through our SMTP server.

It looks as though we may have an incident, because it is not normal for a workstation to send so many emails to external addresses in such short a period of time. This concept reinforces what we discussed earlier: Incidents are based on a deviation from the norm and harm or the attempt to harm. We need to know what is normal, so we can better detect what is abnormal. In this example, sending hundreds of external emails is not normal at all and should be investigated further.

But, what about harm? Is this harming our organization? Yes, it is. It can impact your organization's reputation. In fact, it is common when something like this occurs, the SMTP server(s) of the infected organization can get placed on spam blacklists. This would prevent your users' emails from being received by other organizations, which would undoubtedly have an adverse impact on operations. Further, because a customer of your organization reported the initial anomalous activity, your organization's reputation likely has been impacted in that customer's view.

© 2016 Ed Skoudis, John Strand



Next, let's look at the memory analysis of 192.168.49.144. As stated earlier in the lab, all you need to do to open this file is install Redline and open the Friday.mans file from your C:\tools directory.

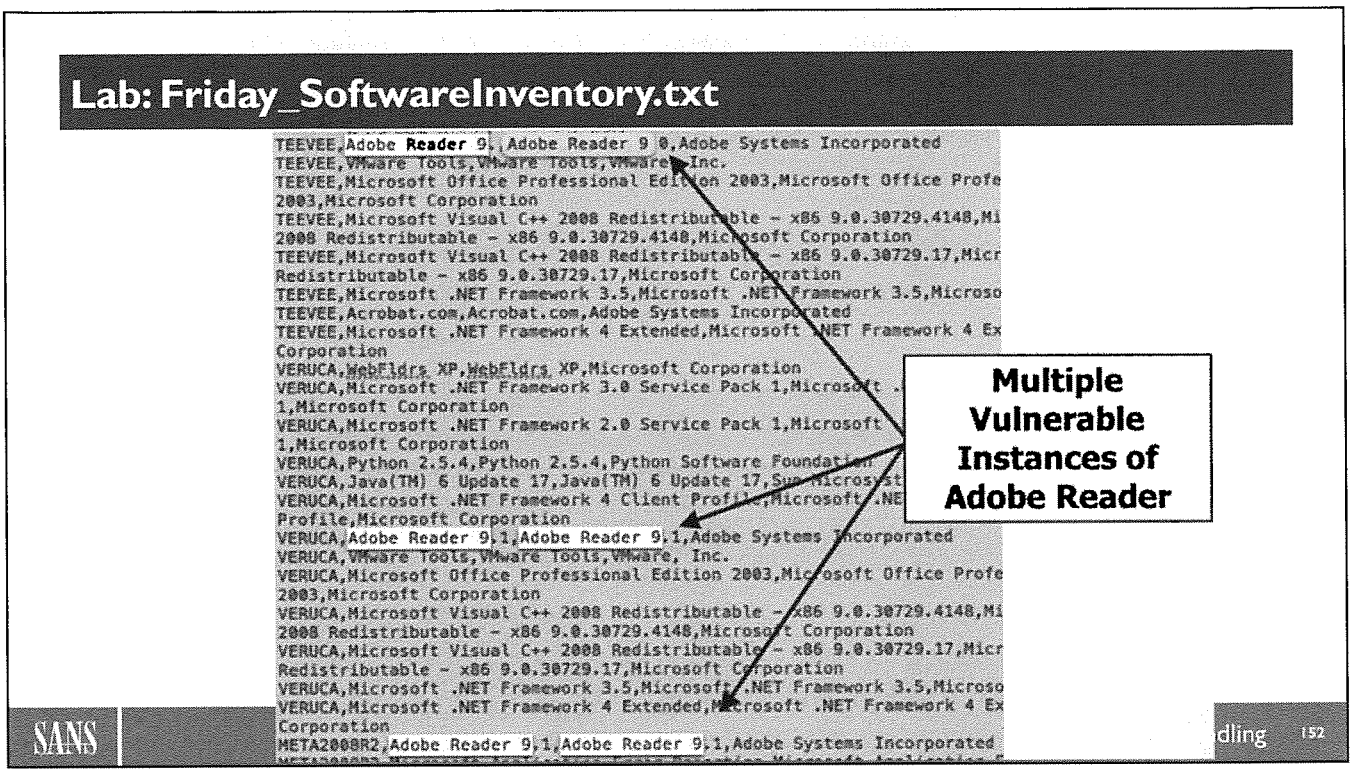
When it starts, you may need to select "I am reviewing a full live memory response or memory image."

Once this file is open, simply click Timeline (Step 0) in the left pane and then scroll until you see red (Step 1) in the right pane. Notice Redline flagged nc.exe. If you remember from the Windows command-line labs earlier, Netcat is a tool that simply reads and writes data over a network connection. However, it can also be used as a backdoor. But, why did it flag one and not the others? The reason it flagged the one nc.exe was because it was spawned from the command shell. That's it. There is no other reason. If you look above Netcat about five lines, you will see hot_pics.exe and above that AcroRd32.exe (Step 2). It appears this system may have been compromised via a PDF exploit that dropped hot_pics.exe on the system. And, it looks like it was launched from Bob's Desktop (Step 3).

But, is that the whole story? Unfortunately, no. We will be doing a deep dive on this exact same memory dump in 504.5. Although Redline is a great tool for quick analysis, Rekall is an outstanding tool for deeper analysis, which we'll use in the 504.5 book.

On this slide, we make an assumption: This system was exploited via a PDF exploit. This may be correct, but it also may be wrong. When we are working incidents, we make a series of assumptions or theories, then see if the evidence supports those theories. In 504.5, we fully test this theory to see if Adobe Reader was truly the root of this evil.

© 2016 Ed Skoudis, John Strand



Now that we know a system is compromised, we need to see if any other systems may be vulnerable in addition to the system sending spam.

What we need is the ability to do a quick snapshot of installed software on the systems in this network. The built-in Windows WMIC command has exactly what we need for this situation.

We can take the following WMIC command to pull a list of installed software:

```
C:\> wmic product get name,version
```

Then we can even log in csv format and run against multiple systems:

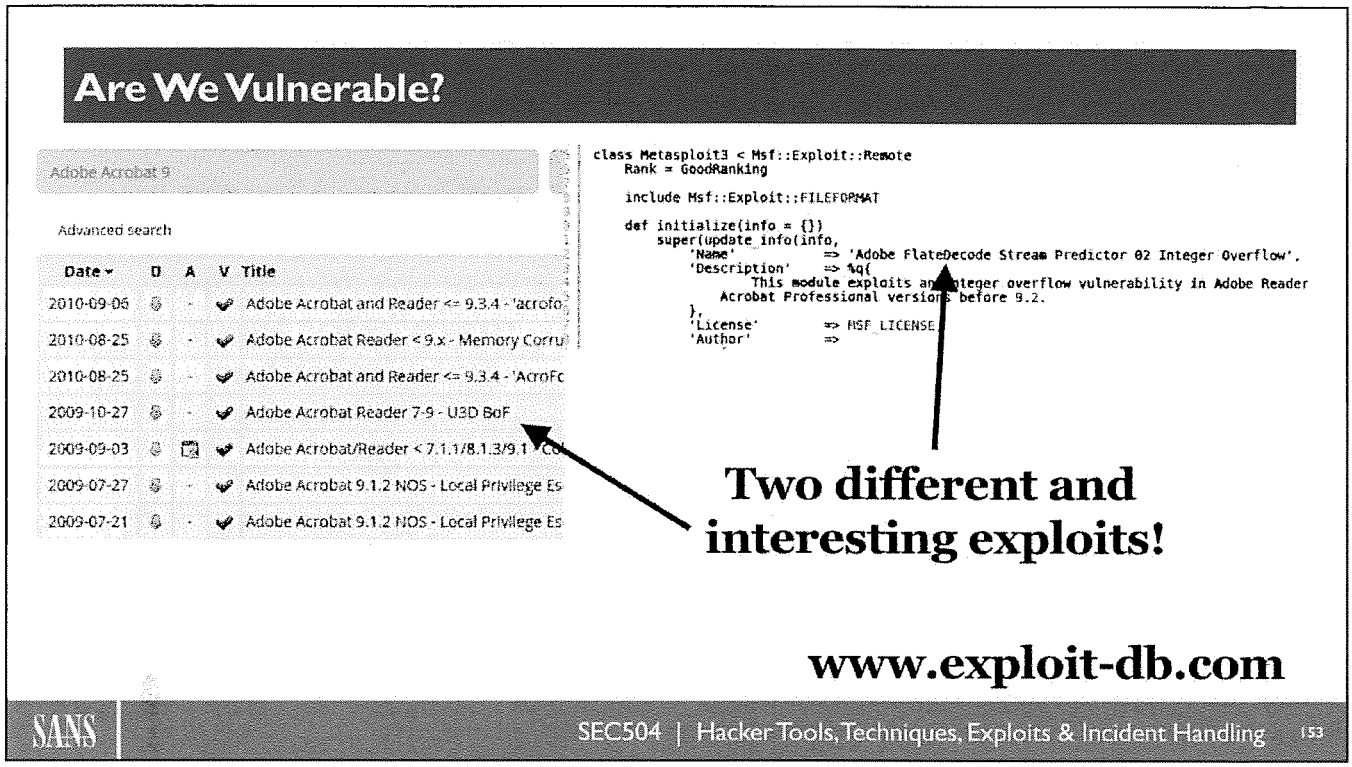
```
C:\> wmic /node:@systems.txt product get description,name,vendor /format:csv
> Friday_SoftwareInventory.txt
```

WMIC's syntax supports an option for /node:@systems.txt, which allows us to run the same WMIC query on multiple systems. Please understand you do not need to do this here. We are simply telling you how the evidence file was created, so you can analyze the evidence.

When we review this file and search for Adobe, you can easily see there are multiple vulnerable systems in this environment.

But, are we sure this could be the root of the evil on 192.168.49.144? Clearly, it is a vulnerability. But now, we just have a theory. We need to test it further to confirm or deny that the attack was launched via Acrobat.

© 2016 Ed Skoudis, John Strand



We know we have some old software. But, are there any published vulnerabilities? It turns out there is a great site for checking vulnerabilities in software: the Exploit Database (www.exploit-db.com). This site's mission is to tie together multiple vulnerability disclosures and notification sites into one place. And, it does a great job of it. For these screenshots we simply searched around exploit-db for a while to find two interesting case examples.

What we did above is take the version of Adobe Reader we discovered on the systems on Friday_SoftwareInventory.txt and checked to see if there were any exploits for that specific version of Adobe Reader. Why did we do this? For two reasons: First, we wanted to see if there were any other versions of Adobe Reader in our environment. Second, the version of Adobe Reader we discovered on the systems in Friday_SoftwareInventory.txt was slightly different from the one on Bob's computer. Bob had version 9.0 and the other systems had 9.1. When we search for 9.0 and 9.1 vulnerabilities, we come across the two specific vulnerabilities boxed in above.
Once again, this lab is meant to be standalone. If you do have an Internet connection, feel free to do a search for Adobe Reader 9.1 and 9.0. However, if you don't have a connection, that is OK. We are simply getting a few details to help search to see if public exploits exist for this flaw.

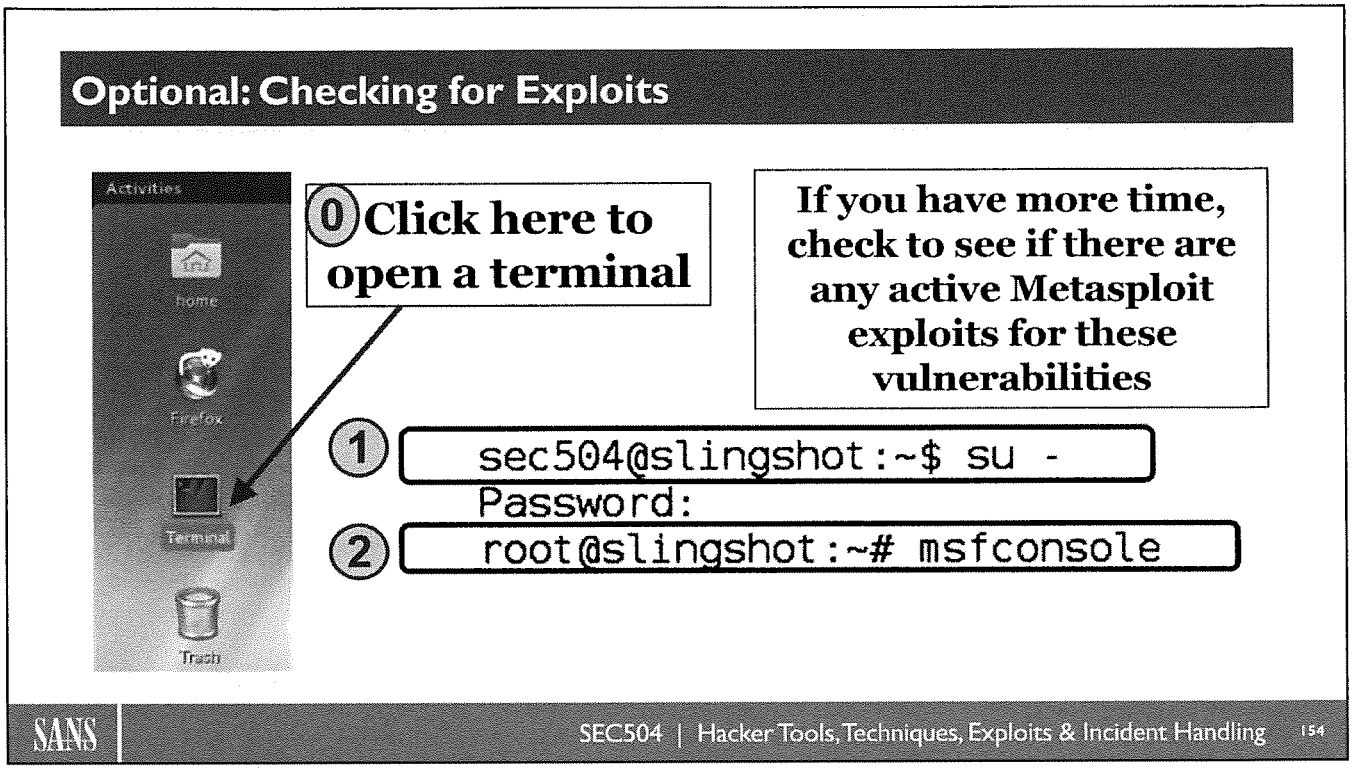
As you can see above, there are three possible vulnerabilities in Adobe Reader that could possibly cause us issues.

But, are there any public exploits available? To determine that, we can research further on the Internet, or we can check to see whether Metasploit, the exploitation framework, has exploits for these flaws.

Do not worry if you do not have an Internet connection. We are simply showing you what you could search for at exploit-db.com. Doing this search on the Internet is not necessary; we are merely using this as a setup for the Metasploit exploit searches on the next slide.

Take a few moments and note U3D and FlateDecode based on the descriptions on the slide above. These topics help us search Metasploit on the next slide. U3D is a code library used for data compression and decompression and FlateDecode is used for encryption and compression of data in PDFs. This concept of flaws in features like this will be a topic of 504.3 when we discuss file-parsing issues.

© 2016 Ed Skoudis, John Strand



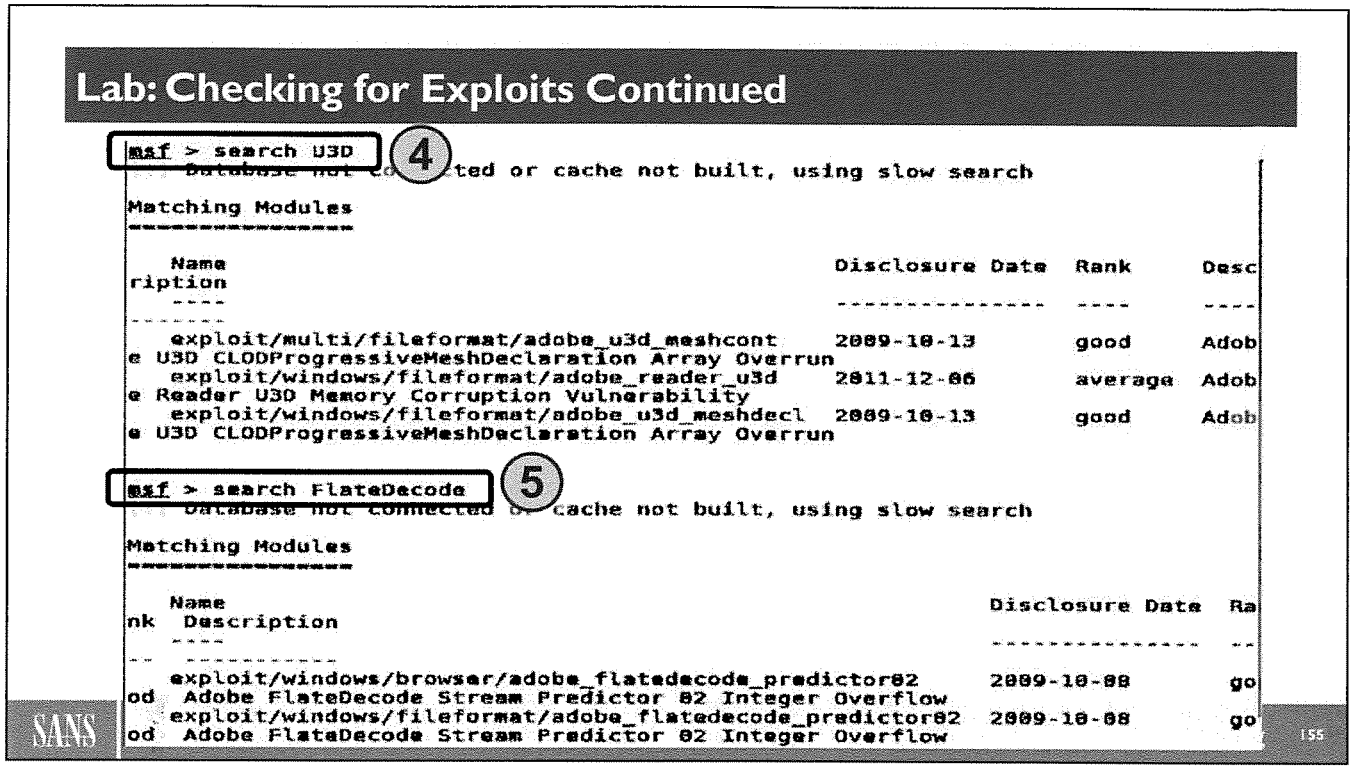
If you have more time, let's fire up our Linux VM and see if there are any exploits for this version of Adobe Reader. We need to do this because, many times, management will not take action until there is proof of an exploit that is easily available to the bad guys.

If you have never run Linux before, don't worry. This will be easy. Also, we have a full Introduction to Linux at the end of this 504.1 book. We are just going to log in and check Metasploit for any active exploits for Adobe Reader 9.1.

First, you need to start your virtual machine. If you have trouble with this or have never used VMware, be sure to ask your instructor for help. Next, we need to log in. The user IDs and passwords to log in are in the Release Notes on the Course USB. As long as you have the USB, you will always have the passwords for the class VM. However, in order to log in for this lab, select "**sec504**" at the login screen and use the password '**sec504**' **without the single quotes** to log in.

Step 0: Once you have logged in, simply select the terminal icon on the Desktop.
Step 1: Become a super user by typing **su** – and entering your login password.
$ **su -**
Step 2: We start Metasploit. Note that it may take a while for it to start. This is perfectly normal; Metasploit is a large program.
msfconsole

© 2016 Ed Skoudis, John Strand



Now, we search for the different vulnerabilities OSVDB listed. Note that we are searching based on a characteristic of each of the different vulnerabilities OSVDB found. Specifically, we look for the vulnerable functions in Adobe Reader 9.1: U3D and FlateDecode.

Step 4: First, we look up U3D, which was a vulnerable function in Adobe Reader 9.1 for which two exploits were released 2009-10-13.

msf > **search U3D**

We see some results. It appears that there are exploits publicly available for these flaws.

Step 5: Finally, we look for FlateDecode. This is a vulnerable function in Adobe Reader 9.1 for which an exploit was released 2009-10-08.

msf > **search FlateDecode**

Note it is "Flate" not "Flat."

Now, we have enough information for management to make well-formed and informed decisions. We have an attack, we have other systems that are vulnerable, and we have active exploits in the wild.

By the way, if you want to know more about Metasploit, we have a full lab dedicated to this great tool in 504.3.

© 2016 Ed Skoudis, John Strand

Lab: Conclusions

- Now that we have reviewed the evidence in this case, we can answer the opening questions:
  1. Have we been compromised?
     - Yes, the e-mail logs are clear that 192.168.49.144 has been spamming a large number of accounts.
  2. Which system has been compromised?
     - 192.168.49.144 is running Netcat, a well-known backdoor.
  3. How was it possibly compromised?
     - Through an Adobe Reader exploit? Possibly. We will need more data.
  4. Who is the user on that system?
     - Bob
  5. Are any other systems at risk?
     - Yes, every other system that was reviewed had an old version of Adobe Reader installed.
  6. What are our containment next steps?
     - Update AV
     - Update Adobe Reader
     - Monitor outbound emails very closely
- There is more to the story. We revisit this case in 504.5 with the Rekall memory analysis lab!
- Like Memory Analysis? Check out SANS Memory Forensics In Depth
  - http://www.sans.org/course/windows-memory-forensics-in-depth

Now that we have reviewed the evidence in this case, we can answer the questions associated with this lab.

1. Have we been compromised?
Yes, the e-mail logs are clear that 192.168.49.144 has been spamming a large number of accounts.

2. Which system has been compromised?
192.168.49.144 is running Netcat, a well-known backdoor.

3. How was it most likely compromised?
The system may have been compromised by a vulnerable Adobe Reader, for which there are several publicly available exploits. However, we need to do a deep dive into the memory dump to confirm. We will do this on day 5 of the class.

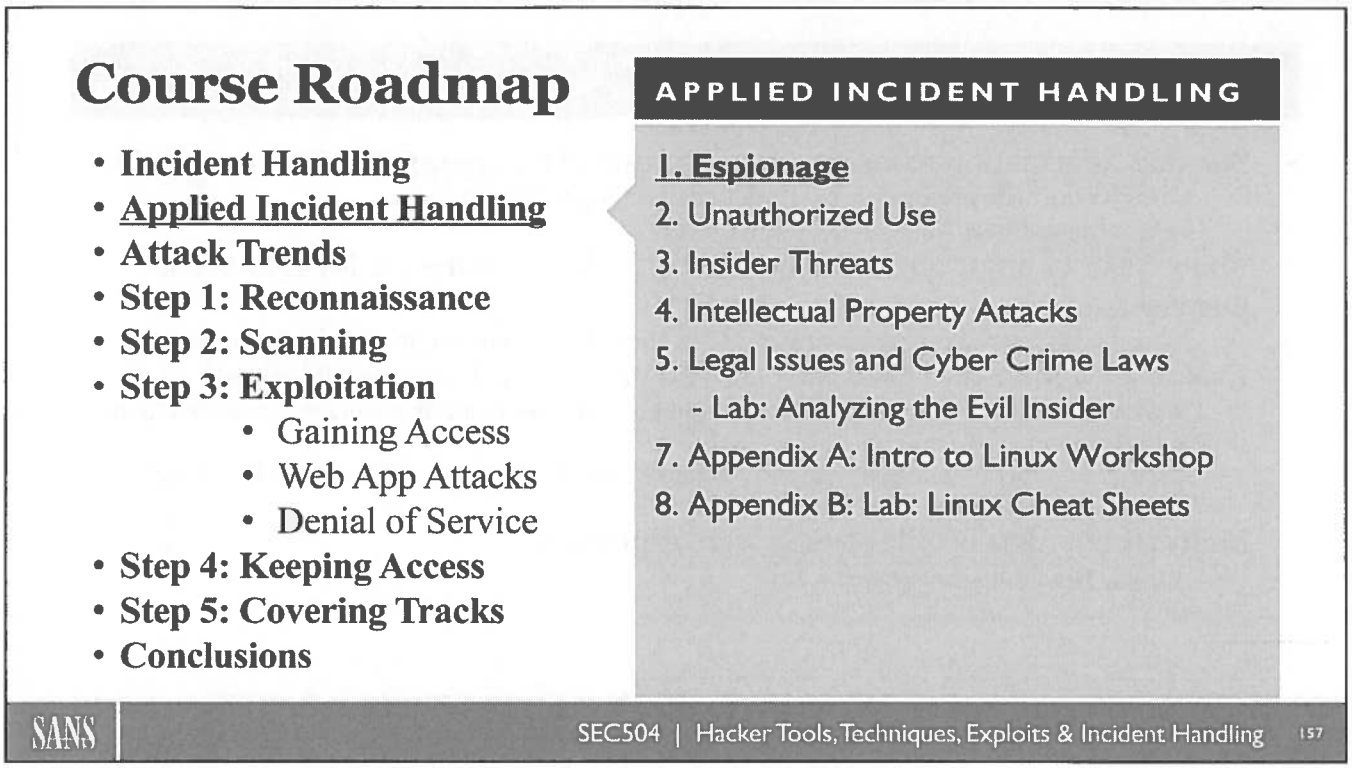
4. Who is the user on that system?
Bob.

5. Are any other systems at risk?
Yes, every other system that was reviewed had an old version of Adobe Reader installed.

6. What are our containment next steps?
Update AV.
Update Adobe Reader.
Monitor outbound emails very closely.

This is the bare minimum amount of information which should be captured when communicating next steps with management. Also, this is not the end of the story. We will be revisiting the compromised system in 504.5 with a very cool memory analysis lab.

© 2016 Ed Skoudis, John Strand



At this point, we completed the six step incident-handling process: Preparation, Identification, Containment, Eradication, Recovery, and Lessons Learned. This is a seasoned process that has worked for a large number of organizations and will work for you. Next, we work on some of the specific types of incidents that we may run into.

We look at several incident types and apply our six-phased process to them. We start with espionage.

© 2016 Ed Skoudis, John Strand

Espionage

- Stealing information to subvert the interests of an organization or government
  - Consider the high-profile Aurora, Titan Rain, and Night Dragon cases
  - Large organizations, high-value information theft
- Many cases of unauthorized access to corporate systems are for espionage purposes
  - Almost every case of espionage prosecuted by the U.S. Government involved a trusted insider
- Espionage and insider criminal cases do not benefit from many helpers
  - Risk of information leak or evidence contamination rises as additional workers are added to the investigation
  - A senior member of management, such as the CIO or Chief Security Officer must be advised, as well as the legal staff
- Many recent high-profile attacks were espionage
  - Aurora, Titan Rain, Net Traveler, RSA

If you are thinking, "Espionage could never happen to *our* organization," think again. Espionage is not limited to governments and the military in any way. The focus of most espionage is not military; it is economic, and this includes the work done by government intelligence agencies.

Businesses routinely are involved in the activities of collecting information about their competition or trying to prevent the competition from getting information about their activities. As long as this is legal, we generally refer to this as competitive intelligence.

As handlers, we are primarily focused on the defensive strategies, of course, but we should take a minute to think about some of the obvious offensive techniques. Some common techniques include:

- Open source searches by your adversaries to see what information is publicly available
- Posing as a customer or potential customer to gain sensitive data
- Hiring critical employees as insiders, in effect working on behalf of your adversaries from the inside of your organization

There have also been a large number of recent high-profile attacks that were motivated by stealing sensitive corporate secret against large multinational companies.

© 2016 Ed Skoudis, John Strand

Espionage Preparation – Target Analysis

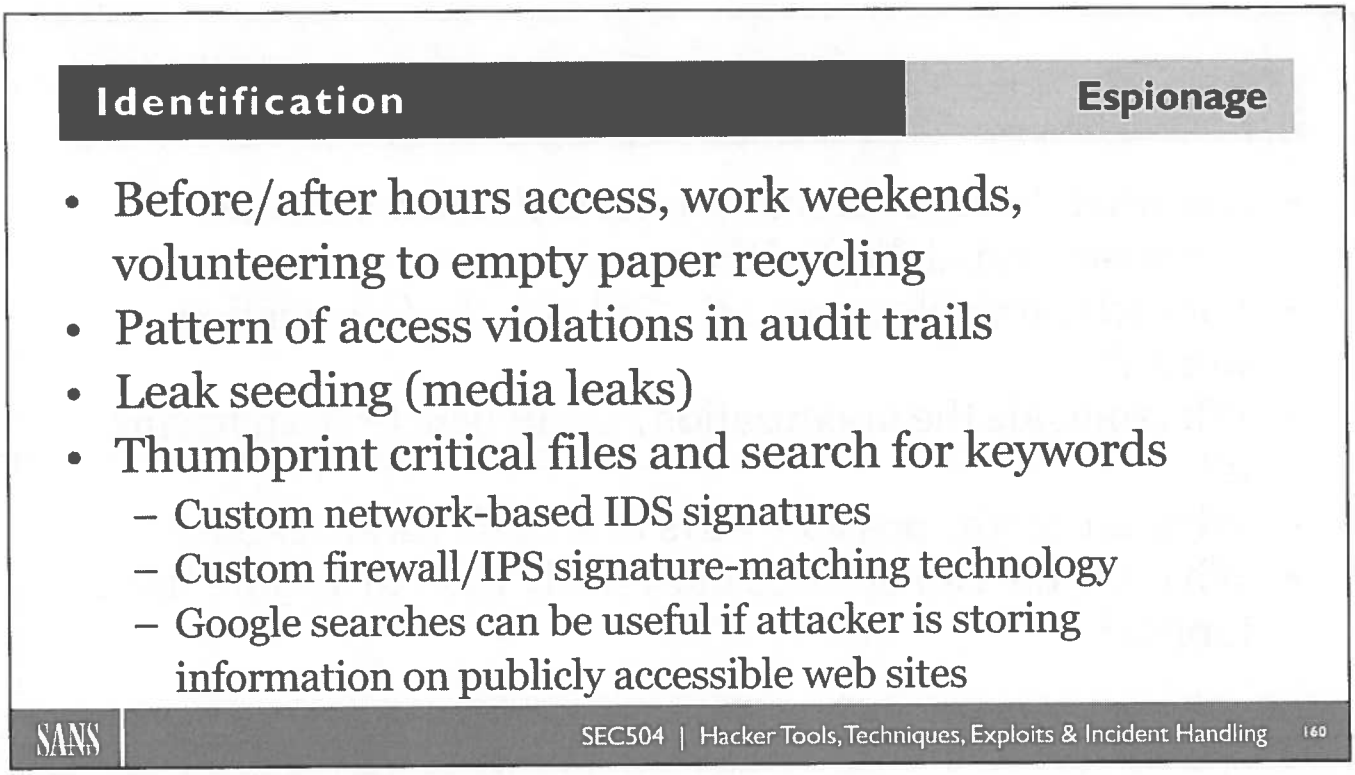
Espionage

- Ask what the most probable targets (information and processing capability) of the activity are
- For each probable target ask, "What is the information worth?"
- Who (outside the organization) might benefit from having it?
- What are all the possible ways to acquire these targets?
- What are the two or three most likely ways to acquire these targets?

KODAK referred to its multi-layer emulsion technology as the crown jewels. What are the jewels that are most valuable in your environment? These are quite likely the target(s).

This is critical; the physical differences between your organization and your competition are probably minimal. Those trade secrets, marketing contacts, business plans, and so forth make all the difference. The odds are fairly high that these crown jewels are the target. By now, you are learning to think like an attacker to some extent. What would you go after if you were interested in plundering your organization? How would you get to this information? Document the results of this exercise and share the information with management to get their input.

© 2016 Ed Skoudis, John Strand



The activity that begins too early before the working day or goes on too late has always been a good indicator. Technology changes, not human nature. Many organizations have been sold out by trusted insiders; the trick is to be alert for this, to work to keep awareness up, and keep running leads to ground.

A great way to thumbprint critical files is to invent an acronym that doesn't actually exist and plant it into the document. Then, if you have content-sensing firewalls, intrusion detection systems, or network-based Intrusion Prevention Systems, they can be set to look for the string. Google searches can also be useful in finding data that has been leaked and placed on the web.

Intentional leaks work well, especially as you start to close in, and this is a standard practice used to hunt down the source of information being released.

© 2016 Ed Skoudis, John Strand

Maximize Data Collection

Espionage

- Ensure that access records of the affected facility are collected and protected
- These may include
  - Records from badge access systems
  - Phone records from your organization's PBX
  - Log books
  - System logs
  - Network logs
  - Surveillance videos
- Collect as much back data as possible
- Make sure you can get access to this type of data when you really need it...
  - ...by asking for it periodically, as a course of doing business

One of the huge challenges is to get some degree of chain of custody when you are dealing with a lot of data and in a large number of formats. This can quickly end up filling a dozen or more copy paper boxes. When possible, take cryptographic hashes of critical log files; these are very small and can be stored in multiple locations. I have even mailed them to trusted friends simply asking them to store the hash or PGP signature of file for me without telling them why. To the extent possible, make every effort to keep the data as pristine as possible.

Also, you want to use some care in the collection of the data; you can get away with one cover story, but at some point, people start to talk. This is why incident scenario training is such a valuable tool. If you are asking for door "badge swipe" records a couple times a year and it is well-known you are training, you not only make sure the logs are being kept and can be retrieved, but when you really need them, it will not arouse anywhere near as much suspicion or interest as it would if no one had ever asked for them before.

© 2016 Ed Skoudis, John Strand

Purposely Deceiving the Attackers — Espionage

- If an outsider is collecting the information, you may be able to provide erroneous information and actually benefit from the incident
- If you suspect the information is being collected and distributed by an insider, this is less likely to work
  - However, the technique can be used to pinpoint the insider
  - Make up a fake activity called "Project XYZ" or a bogus bid for a client
  - Configure network-based IDS and/or antivirus tool with custom signatures to look for this fake data

You may also be able to use erroneous or misleading information to detect that a leak exists. This is a classic trick and can sometimes not only help track down the insider, but if you are really clever, you can use this to your advantage.

One of the most difficult cases I have worked on involved contract bids, where there was some reason to think that an insider, probably a system administrator, was accessing information on the bid. In this case, we constructed a completely fabricated bid (that turns out to be a lot of work, by the way), and really kept up the act, making changes as the bid deadline approached, sending information from the boss to the folks making up the bid. All the while, the real bid was being prepared by a different team composed of a couple individuals that were supposedly "on leave." The idea was to feed the other side enough wrong information to really screw them up. It didn't work; probably there was some error in the fake bid that tipped our hand. Either the other side was actually that sharp that they could spot the fake or perhaps we were dealing with a second person that was able to observe what we were doing. This is a hard sport!

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

APPLIED INCIDENT HANDLING

1. Espionage
2. **Unauthorized Use**
3. Insider Threats
4. Intellectual Property Attacks
5. Legal Issues and Cyber Crime Laws
   - Lab: Analyzing the Evil Insider
7. Appendix A: Intro to Linux Workshop
8. Appendix B: Lab: Linux Cheat Sheets

For incidents involving unauthorized use, the attacker is allowed normal access in the course of doing business. However, the attacker abuses this access, using it in an unauthorized fashion.

© 2016 Ed Skoudis, John Strand

Unauthorized Use

- For unauthorized use, the user is allowed normal access, but is abusing it
- Numerous possible categories...
- ...But let's focus on two areas that incident handlers are frequently called upon to support:
  - E-mail problems
  - Inappropriate web surfing

Unauthorized use is something that every handler is likely to face. We deal with a couple common problems: unsolicited, non-spam e-mail and inappropriate use of computing resources. As we look at this section, keep in mind that you are an incident handler, not a cop, not a lawyer, not your organization's Human Resources department. Don't cross the line and do their jobs for them. On the other hand, you are likely to be coming from a technical perspective. Who better then to assess the situation and collect, protect, and analyze any evidence than the handler?

© 2016 Ed Skoudis, John Strand

E-Mail – Gathering Evidence

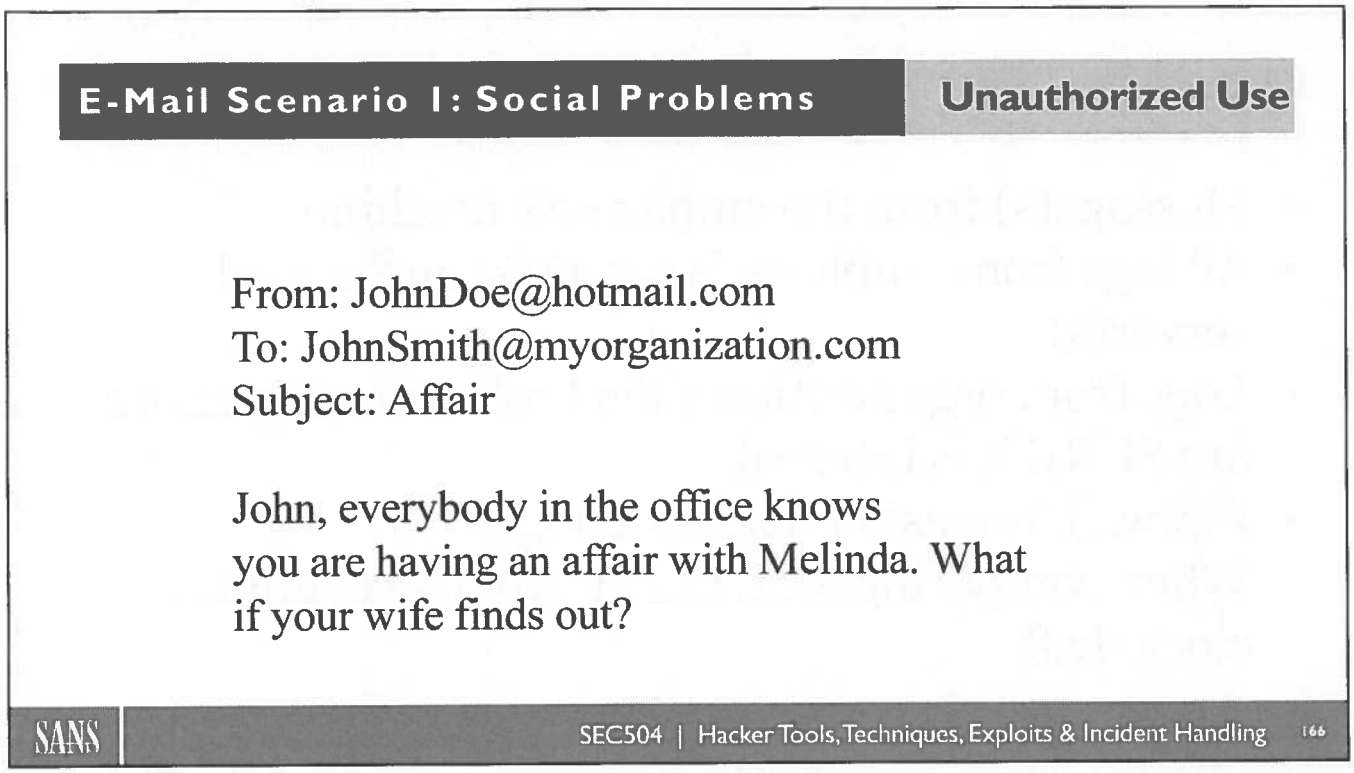
Unauthorized Use

- Message(s) from the employee's machine
- All logs from employee's server(s) and e-mail server(s)
- Logs from organization's mail relay(s), even if you are SURE it is internal
- Firewall/intrusion detection logs
- When comparing logs, be certain to account for clock drift

Keep in mind that you may face situations where the mail appears to be coming in externally; for example, it is coming from the Internet, but it could still be someone internally sending it. The simplest example of this is when an employee surfs to Hotmail to send the note into the organization. If you have decent monitoring, this is easy to run to ground. If he uses a telephone and call up to an ISP to send the mail, you may have to pull phone records or work with law enforcement to subpoena records from the ISP.

Many log files are perishable. As soon as you know you may need to collect evidence, you probably should. Have I mentioned that it is a good idea to SHA-1 and MD5 hash logs after they are closed? This point about perishable is important: Many times, the employee that is receiving the offensive mail doesn't say anything until he has received 15 or so over a 6-week period. Then, he hands the stack to his manager, who doesn't know what to do, so she sits on it a week before giving it to HR, and HR waits two days until the manager calls back and then it calls in a handler. Remember from the first part of this course, we were talking about awareness and knowing how to contact your organization's incident-handling capability? Situations like this are where the rubber meets the road. If you don't know there is a problem for 6 weeks, it may not be possible to collect all the logs needed to do a complete investigation.

© 2016 Ed Skoudis, John Strand



Handlers are not responsible for mending broken hearts, but they may have to deal with them. A common situation is for a female employee to receive unwanted e-mail. It is usually pretty easy to determine the source in these cases; after all, the prospective suitor wants to be noticed and found. Once the data is in, this is an HR thing, not a handler issue.

Domestic disharmony is also usually pretty easy to run to ground; after all, the source will be someone close to the victim. Take a look at this message; who is on your shortlist for people that might have sent this? John Smith's spouse and her close friends and relatives. If any of them work in the same organization, that could be a clue. For some reason, Hotmail has been the most often used service for insider sent problem e-mail messages.

Sorry to harp on how the initial data is often misleading, but I worked a case in Colorado where malicious code that damaged five computers was sent into the organization from an account with the name of a female employee@hotmail. By the time I was alerted, they had restored all the operating systems on the computers, which put the best evidence in doubt. They had also begun the inquisition on the female employee. We were able to pull the mail, firewall, and IDS records, but this facility had a tradition of people going to Hotmail during the day, so we had about 20 leads. We actually were able to solve this case and get a conviction, but it was by pure dumb luck, not by something we were able to do; it was a 15 year-old kid who bragged about what he did.

© 2016 Ed Skoudis, John Strand

E-Mail Scenario 2: Phishing

Unauthorized Use

From: security@bigbank.com
To: victim@yourISP.net
Subject: Critical Changes to Your Online BigBank Account

Dear BigBank valued customer,
In order to service you better, we have made some changes to the way you access your BigBank Online Account. From now on, access will be managed by BigBank Online Management Center. Please note that BigBank Online Management Center will be the only way you can access your account. Click on the following link to access your account:
www.bigbank.com/login.asp

You can report phishing to the bank (or other org that appeared to send e-mail), the ISP, and www.antiphishing.org

This e-mail represents a phishing attack. The perpetrators are trying to harvest account information from unsuspecting users by setting up a website that poses as the actual bank's site. Then, with an e-mail blast often carried by worm-infected systems (see the previous slide), they trick users into surfing to the attacker's site. You should report phishing to the bank, the ISP, and www.antiphishing.org.

The links included in phishing e-mails are actually accessing the attacker's site, but trick a user in any one of a variety of ways. Often, the attackers use an <A HREF> tag to display certain text on an HTML-enabled e-mail reader screen, with the link actually pointing somewhere else.

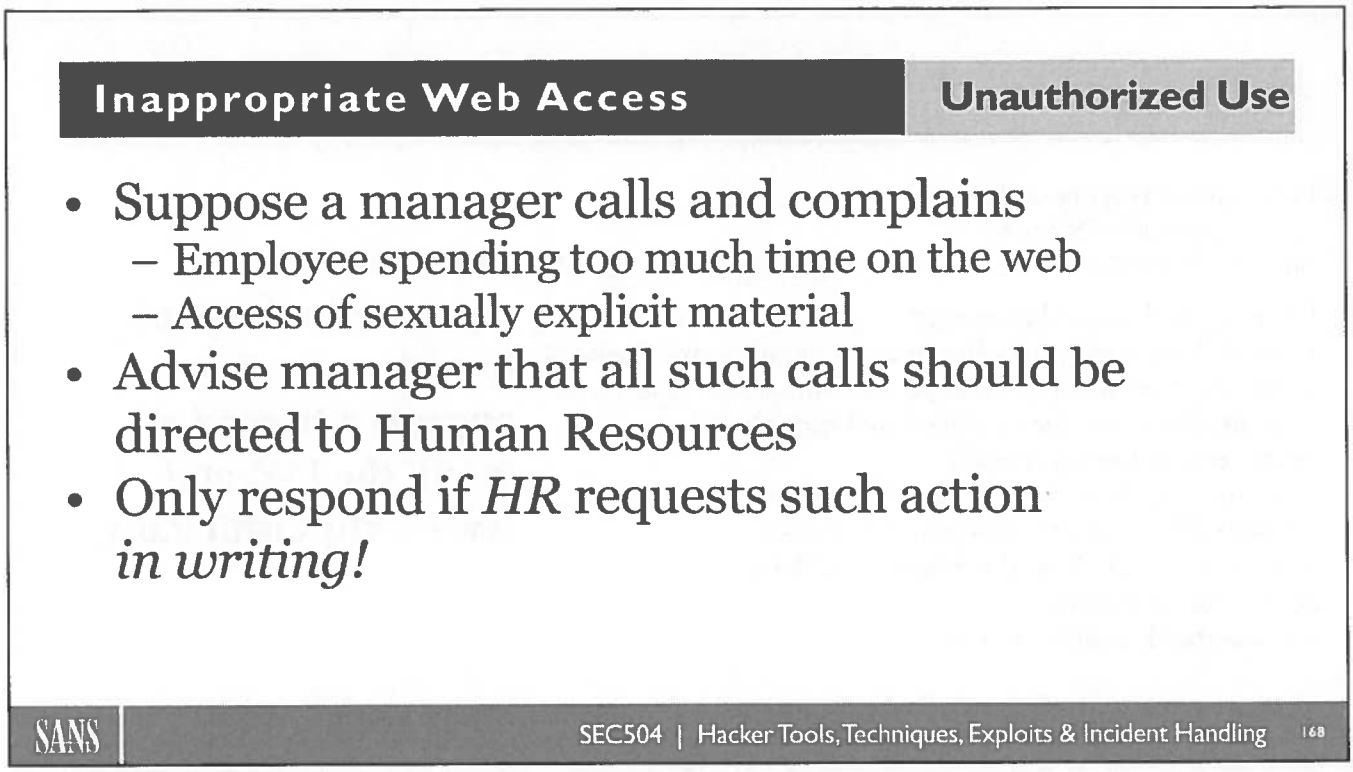
First, and perhaps most simply, the attacker could simply dupe the user by creating a link that displays the text "www.goodwebsite.org" but really links to an evil site. To achieve this, the attacker could compose a link like the following and embed it in an e-mail or on a web site:

```
<A HREF="http://www.evilwebsite.org">www.goodwebsite.org</A><p>
```

The browser screen will merely show a hot-link labeled www.goodwebsite.org. When a user clicks it, however, the user will be directed to www.evilwebsite.org. Browser history files, proxy logs, and filters, however, will not be tricked by this mechanism at all, because the full evil URL is still sent in the HTTP request, without any obscurity. This technique is designed to fool human users. Of course, although this form of obfuscation can be readily detected by viewing the source HTML, it will still trick many victims and is commonly utilized in phishing schemes.

More subtle methods of disguising URLs can be achieved by combining the above tactic with a different encoding scheme for the evil web site URL. The vast majority of browsers today support encoding URLs in a hex representation of ASCII or in Unicode (a 16-bit charac set designed to represent more characters).

© 2016 Ed Skoudis, John Strand



Unauthorized use comes up from time to time. An incident handler may receive a call from a manager saying, "I'm concerned about Bill Smith, his work appears to be off, can you

A) Access his e-mail

B) Use the intrusion detection system to track his activities

C) Copy his hard drive while he is away from his desk

D) All of the above"

Your answer, even in an organization with warning banners and no presumption of privacy, should be NO! If the same call comes in from Human Resources, and you have written policy on when your organization might do any of the above, then ask them to confirm their request in writing and you may have to do these things because you are an expert on collecting and assessing evidence.

It is one thing to randomly monitor for certain strings, "SECRET", "CONFIDENTIAL", even "Supermodels" and another thing entirely to use your skills to target an individual. Make sure that you are on firm, written, legal ground.

© 2016 Ed Skoudis, John Strand

Corporate Access of Sexually Explicit Websites

Unauthorized Use

- Sexually explicit web access is a major problem for many organizations
- Such material is considered inappropriate because
  - It could be a factor in a sexual-harassment case
  - It could embarrass your organization
  - These sites sometimes distribute spyware and other forms of malicious code
- Our jobs as incident handlers involves helping our organizations minimize damage from the misuse of computer systems...
- ...so we sometimes are called to get involved with this type of situation

You know the story about the elephant on the table? There is an elephant on the dining room table, but no one will talk about it. This is used to illustrate the issues with a substance abuser in a family. Everybody sees it, but no one wants to talk about it.

Sexually explicit access is one of the biggest money vectors on the Internet. Yet, no one wants to talk about it. A lot of your organization's time is lost. Further, if countermeasures are not applied, your organization could be in for a major lawsuit. That said, incident handlers are not responsible for the organization's policy. Senior management manages risk and, in the final analysis, this is simply one more risk-management decision.

One thing you want to establish from a policy perspective is what to do if you happen to run into images that are illegal, such as child pornography. Do you call law enforcement? Probably; there is a better than even chance that they already know those images are in your facility. Speaking of possessing illegal materials, if your organization has a public FTP server, be sure and check it closely from time to time for hidden directories. The best place to store illegal files is on someone else's computer.

© 2016 Ed Skoudis, John Strand

Preventing Problems with Filtering Web Proxies — Unauthorized Use

- If such activity is suspected, the wisest choice is to implement proxy countermeasures to block access to such sites
  - WebSense
  - Blue Coat
  - Numerous others
- This is not foolproof, but it works for flow reduction
  - By stopping 90% of the problem, it allows handlers to focus on the remaining 10% and other sensitive issues
- Also, it shows the organization's culture does not condone such activity

Nothing beats NetNanny-style filtering software to keep your organization out of hot water. For one thing, you just can't argue with a proxy about what is appropriate or not; it just does its thing and people seem to accept that. Also, you really don't want a sexual-harassment suit; it takes a lot of time and money to deal with, and your organization will get a lot of negative press.

You should encourage your organization to filter out unwanted web sites using a web-filtering tool, such as WebSense, Blue Coat, or others.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

APPLIED INCIDENT HANDLING

1. Espionage
2. Unauthorized Use
3. **Insider Threats**
4. Intellectual Property Attacks
5. Legal Issues and Cyber Crime Laws
   - Lab: Analyzing the Evil Insider
7. Appendix A: Intro to Linux Workshop
8. Appendix B: Lab: Linux Cheat Sheets

This section covers the classifications of insider threats, how to identify potential insider threats, and how insider threats can be prevented.

© 2016 Ed Skoudis, John Strand

What Is an "Insider" Threat?

Insider Threat

- Simply stated, a threat from an entity with access to your data
  - An employee
    - Including contract and temporary employees
  - A business partner
    - Someone that has legitimate access, but is not an employee
  - Such attackers usually have valid credentials and knowledge of the environment and its business practices

What is an insider threat? Simply stated, it's a threat from an entity with access to your data. This class deals with employee and business partner threats.

A well-intentioned employee makes mistakes that allows proprietary information to leak or access to your proprietary data.

In the physical realm, this could be an employee holding the door into a secure area for the person carrying lots of books or papers. Or someone that states she simply forgot her badge.

In the cyber realm, this could be an employee at a help desk being social engineered out of an ID/password combination or a trusting sole that gives their ID/password combination to another.

The disgruntled employee could be someone who believes he may have been overlooked for a promotion, feels he is better than others and have a need to show it.

The unnoticed employee (a.k.a. the secret thief) is probably the most serious threat you have to your infrastructure. This could be someone operating for a competitor, gathering data for sale to the highest bidder, or someone working for a foreign government.

One point I want to make is that when I use the term "employee," I include employees at all levels (even executives can be suspect), contract, and temporary employees.

© 2016 Ed Skoudis, John Strand

Handling the Insider Case – Warning Banners (AGAIN!)

Insider Threat

- Required if monitoring may lead to disciplinary action or prosecution
- Warning banner must advise the user
  - That access to the system is limited to company-authorized activity
  - That any attempted or unauthorized access, use, or modification is prohibited
  - That unauthorized users may face criminal or civil penalties
  - The use of the system may be monitored and recorded
  - If the monitoring reveals possible evidence of criminal activity, the company can provide the records to law enforcement
- Verify wording with legal staff, and be careful with European Data Privacy Directives

Before any electronic detection of insider threats can begin, you must ensure that employees are aware of your monitoring policy.

Should your monitoring uncover any illegal activity it will be hard, if not impossible, to pursue any criminal or civil charges without proper warning screens on the front doors of your systems. It is important that your company displays a warning message at each login point. This includes all points of remote access, be it VPN, SSH, FTP, dial-up, and all internal sign-on locations. This includes all single-user and shared terminals. This warning message must cover the following five points:

1) It must advise the user that access to the system is limited to company-authorized activity
2) That any attempted or unauthorized access, use, or modification is prohibited
3) That unauthorized users may face criminal or civil penalties
4) The use of the system may be monitored and recorded
5) If the monitoring reveals possible evidence of criminal activity, the company can provide the records to law enforcement

Make sure your legal staff reviews and approves of this wording in writing. That way, they'll support you if a case should ever be challenged. Also, be careful not to run afoul of the European Data Privacy Directives.

© 2016 Ed Skoudis, John Strand

Identification of Insider Activity

Insider Threat

- Gathering intelligence on system activity
  - What web sites and FTP sites are being visited?
    - Hacking tools , encryption tools, steganography software, free web-based e-mail sites
- Monitor message boards for posted financial or merger information
- Gathering intelligence on your employees' activities
- General searches on the Internet are acceptable
  - Casting a wide net is fine
  - But targeting a particular employee should only be done with written HR approval!

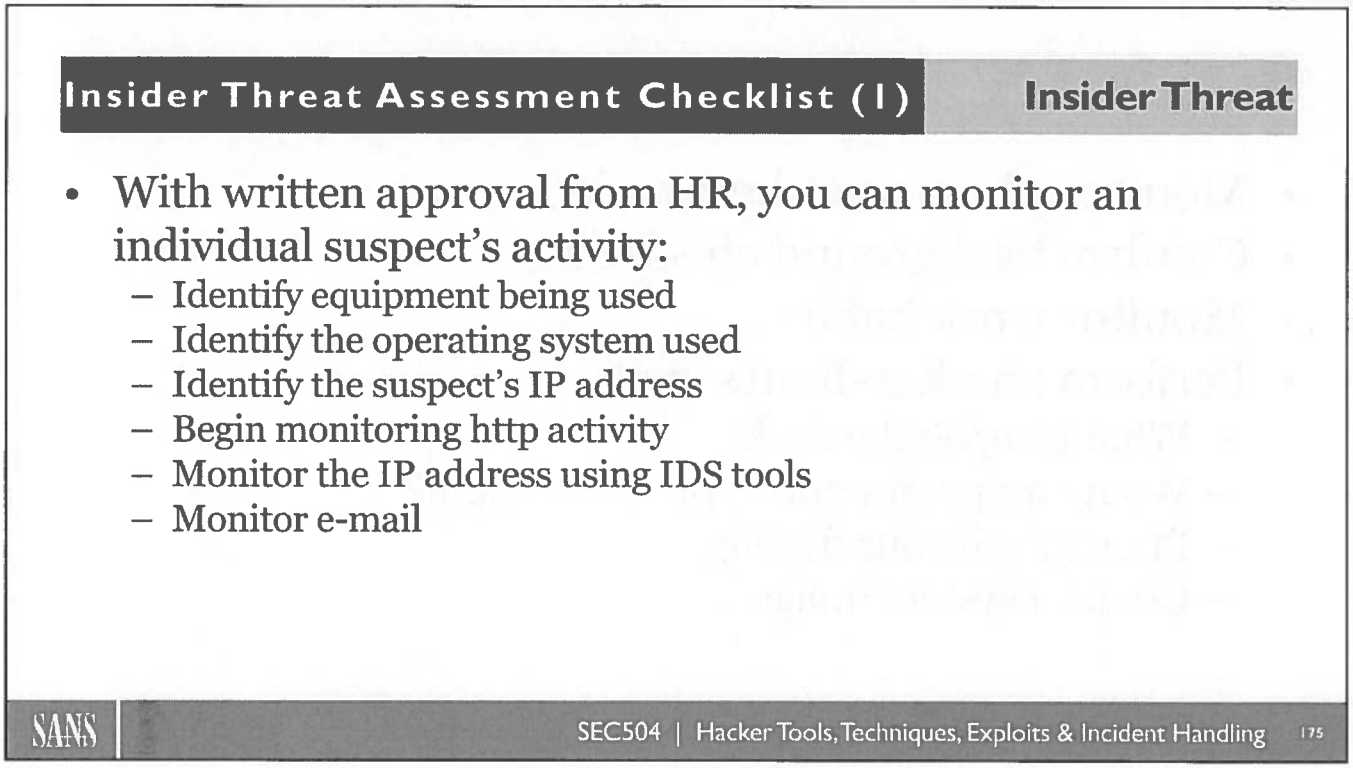
Now that you have warned your employees that they may be monitored, you can begin looking for threats.

The best way to identify insider activity is to gather intelligence through proactive scanning and monitoring. You should always be alert for anomalies, and keep copious records of those anomalies. These records assist you in identifying a threat to the business.

Monitor message boards for posted financial or merger information.

Detection of anomalies can be accomplished by two means: intelligence gathering on your systems and intelligence gathering on your employees' activities. Although no one wants to live in the world of Big Brother, it is important the activity be logged for accountability, company protection, and possible legal action.

© 2016 Ed Skoudis, John Strand



After getting explicit approval from HR, you can start monitoring a specific employee's actions. Make note of the following items:

Equipment identification

A laptop that the suspect takes home or a desktop PC. Is a modem attached? How about a wireless access point? Are there other wireless technologies, such as EVDO, in use?

Operating system identification

This helps you identify the tools to use.

Identify IP address

If dynamic, set to a static IP so you can more easily track the system's activities.

HTTP activity

Are the intranet and Internet sites visited pertinent to the suspect's duties? Are web-based e-mail services being used?

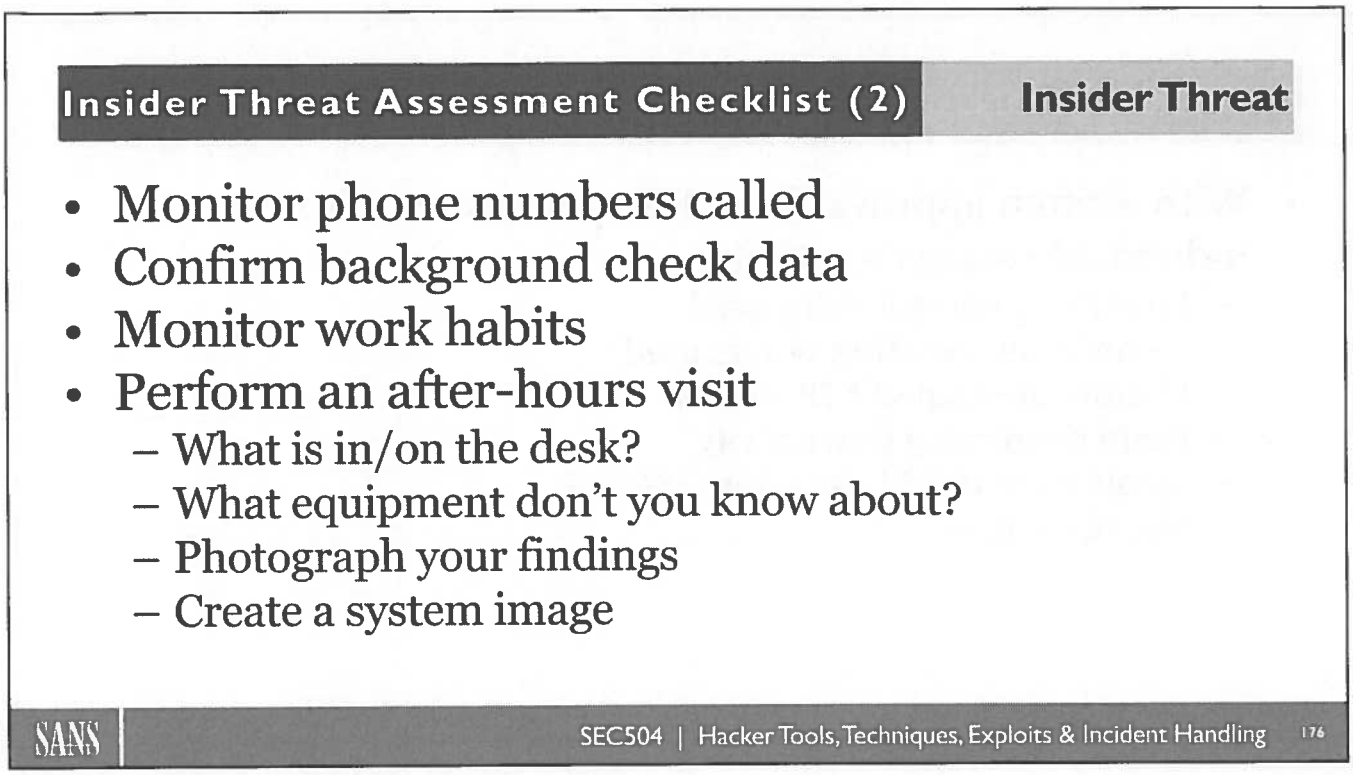
IDS use

Monitor the traffic to/from the IP to identify inbound and outbound traffic.

Monitor e-mail

Grab copies of e-mail to and from the suspect. Be particularly careful to view mail off-line so that you do not send an auto-reply to the message originator.

© 2016 Ed Skoudis, John Strand



Monitor telephone calls made and received

These are the numbers dialed, not the conversations themselves. Look for patterns.

Confirm background check data

Ensure a background check was performed. Review the data so you can better understand the suspect. If one was not completed, consider a background check, but realize that a full check may take several weeks.

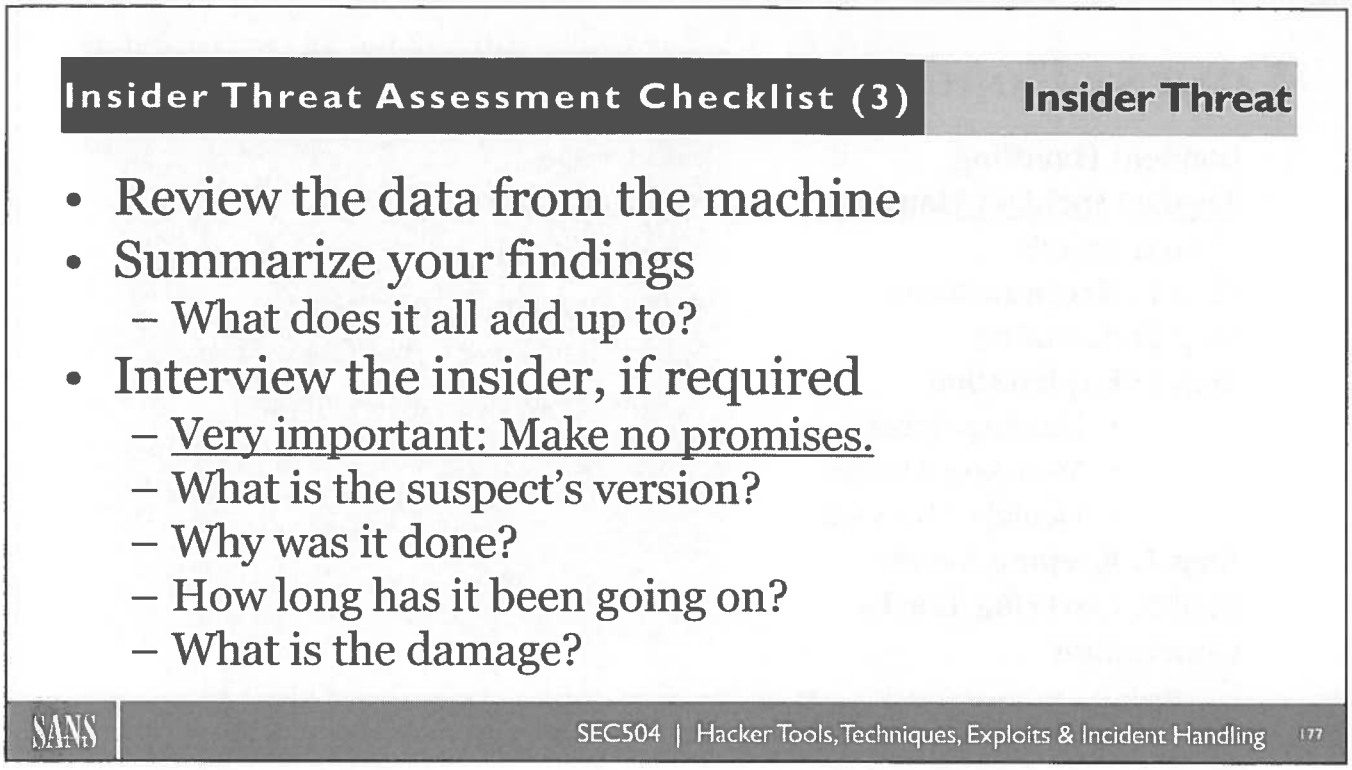
Monitor work habits

Is the suspect working late into the evening? Or working from remote locations more often?

After-hours visit

What is the suspect storing in his or her desk? Look for items that don't belong. If possible, create an image of the suspect's system. An image using some type of imaging software, such as dd, Ghost, Snapback, or Safe Back is the preferred method if civil or criminal prosecution is possible.

© 2016 Ed Skoudis, John Strand



Review the data

The goal is to review not only allocated file space, but deleted files, slack space, swap space, and so on. Tools such as Expert Witness allow you to search the image file for specific file types and character strings. This enables you to determine if the suspect had accessed data without authorization and provide you insight into what else the suspect may be doing. A search for hacking tools, log files, and such helps you identify how the suspect is accessing data.

Summarize your findings

Determine what all the data you have gathered adds up to. Does it look like there is an actual threat or is it a perceived threat? Actual threats need to be eliminated and possibly prosecuted. Perceived threats, someone exhibiting the trigger signs noted earlier, can be transferred and/or offered employee assistance.

Interview the suspect

Very important. Make no promises. This should be viewed as a business meeting where a specific agenda must be completed before the meeting ends. Your goal is to get the suspect's version of what was done. Be sure to answer the "what, where, when, why, and how" of the activity.

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

APPLIED INCIDENT HANDLING

1. Espionage
2. Unauthorized Use
3. Insider Threats
4. **Intellectual Property Attacks**
5. Legal Issues and Cyber Crime Laws
   - Lab: Analyzing the Evil Insider
7. Appendix A: Intro to Linux Workshop
8. Appendix B: Lab: Linux Cheat Sheets

In the information age, Intellectual Property is of major importance. We're going to focus on the legal issues surrounding this increasingly important topic for incident handlers.

© 2016 Ed Skoudis, John Strand

In a Competitive Market, Intellectual Property Dominates

Intellectual Property

- Most players have about the same tooling and capability
  - Coke could produce Pepsi
  - Chevron could produce Exxon
  - But, they would require appropriate Intellectual Property
- Intellectual Property from brand management to the "secret formula" is the primary distinction between two competitors
- Intellectual Property does and will come under continuous attack, and such attacks are rarely detected or reacted to

Think about it: The differences between Pepsi and Coke are mostly Intellectual Property issues! The products may be similar, but the Intellectual Property defines these different businesses. Brand management, distribution, and even competitive intelligence are what really matters. Not the ingredients of the product. Intellectual Property reigns supreme in the Information Age. Therefore, it is a juicy target for an attacker.

© 2016 Ed Skoudis, John Strand

Crown Jewels — Intellectual Property

- Patents
  - Protect inventions and innovations
- Copyrights
  - Protect works/content, particular expressions of ideas
- Trademarks, Servicemarks
  - Protect brands, whether icons, phrases, or sounds
- Trade Secrets
  - Protect sensitive information required for your business...
  - ...which you have taken reasonable precautions to secure

Here are the components of Intellectual Property that could come under attack. We'll explore each of them in more detail in the next several slides.

You, or someone in your organization, should survey the critical Intellectual Property belonging to your organization, using the above elements as a guide.

© 2016 Ed Skoudis, John Strand

Incident Handling for Intellectual Property Cases

Intellectual Property

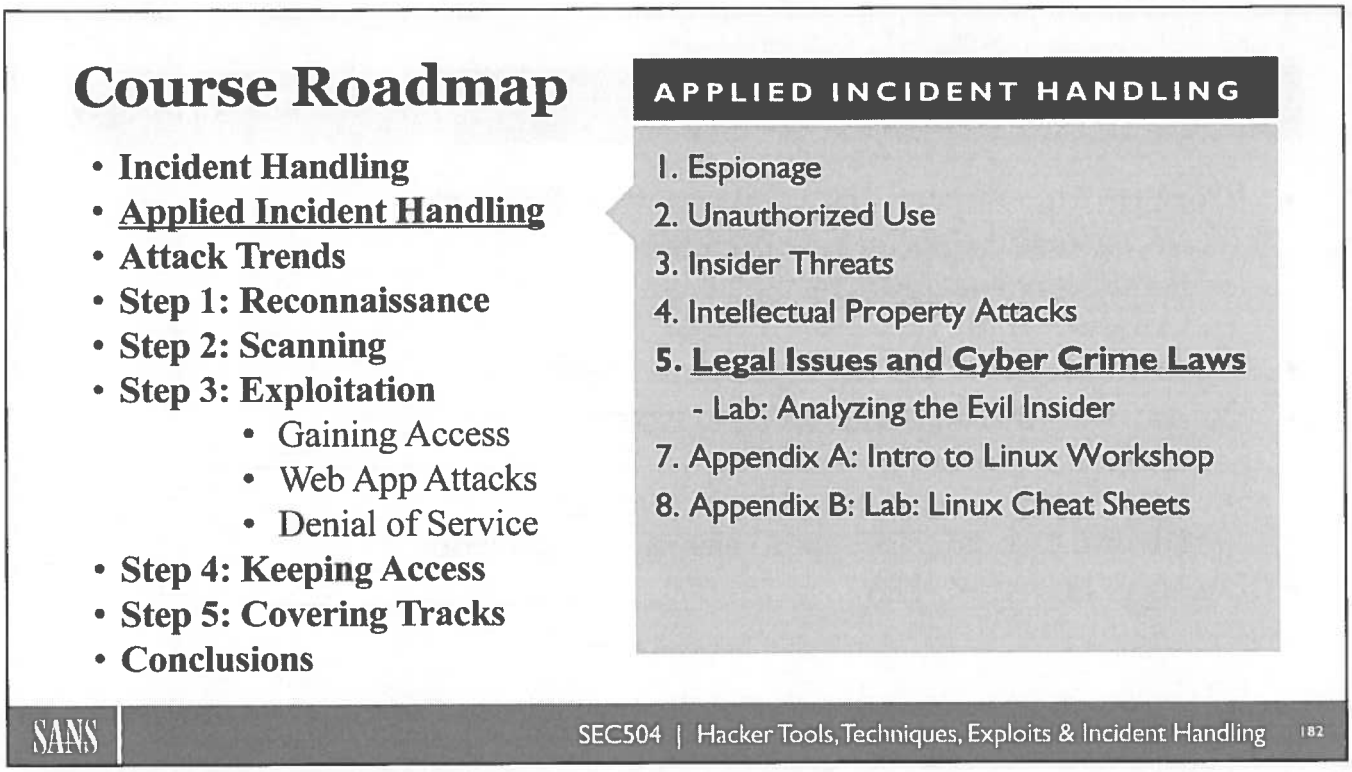
- Preparation: Survey your Intellectual Property
  - Do you have marks for your brands?
  - Is your core material copyrighted?
  - Can you identify your trade secrets?
- Identification: Look for leaks and theft
- Containment: Criminal or civil case?
  - Work with lawyers to decide
  - For criminal case, contact law enforcement
  - For civil case, lawyers issue a cease-and-desist letter
- Lessons Learned -NA

To prepare for an attack against your Intellectual Property, make sure you survey and inventory your Intellectual Property. Work with your legal and PR teams to make sure you understand the important Intellectual Property assets of your organization. Register all appropriate patents, marks, and copyrights. Also, identify your trade secrets and make sure they are adequately protected.

For Intellectual Property attacks that involve publicly disclosing information, you can include a defining element in your critical data that makes it far more searchable. Some attacks are hard to detect (for instance, a counterfeit mark). The most common problem with counterfeit marks would be a mark that is used illegally in another country where there are less protections and these products are used there or imported back to the U.S.

For a particularly important violation of your copyrights or other intellectual property, you may want to file a criminal case to the government for prosecution. However, like the other law-enforcement discussions in the incident-handling training, we find that you have to follow through and build relationships or nothing is likely to happen. So, get to know your local federal law-enforcement officials. They can support you if you go down the path of criminal prosecution. If you and your attorneys decide to tackle the issue in civil court, you will likely initially send a cease-and-desist letter, followed by formally filing a lawsuit if the infringement doesn't stop.

© 2016 Ed Skoudis, John Strand



Now, we move on to Law, Crime, and Evidence. Let's look at the legal issues surrounding incident handling and see how they impact the incident handler's job.

© 2016 Ed Skoudis, John Strand

Country-Specific Cyber Crime Laws

- In most countries, computer crime falls into two arenas
  - Traditional crimes facilitated by a computer (used for storage and analysis)
  - Crimes in which the computer is the target (exploitation, information theft, denial of service)
- Here is a great portal for U.S. cyber crime laws
  - http://www.justice.gov/criminal/cybercrime/
- The texts of the dominant cyber crime laws of over 40 countries (including Norway, Mexico, India, China, and Israel) have been gathered together at http://www.mosstingrett.no/info/legal.html#countries
- Always incorporate your organization's legal department into any incident or interaction with law enforcement

Generally speaking, in most countries, computer crimes fall into two arenas: crimes in which the computer was used to facilitate traditional criminal activity (acting as a storage or analysis device for the criminal who might be dealing in drugs, participating in organized crime, trading child pornography, or engaging in terrorist activities), and crimes in which computers are the target of the attack (where criminals break into, steal information from, or crash computers). Some criminal activity fits into both arenas at the same time.

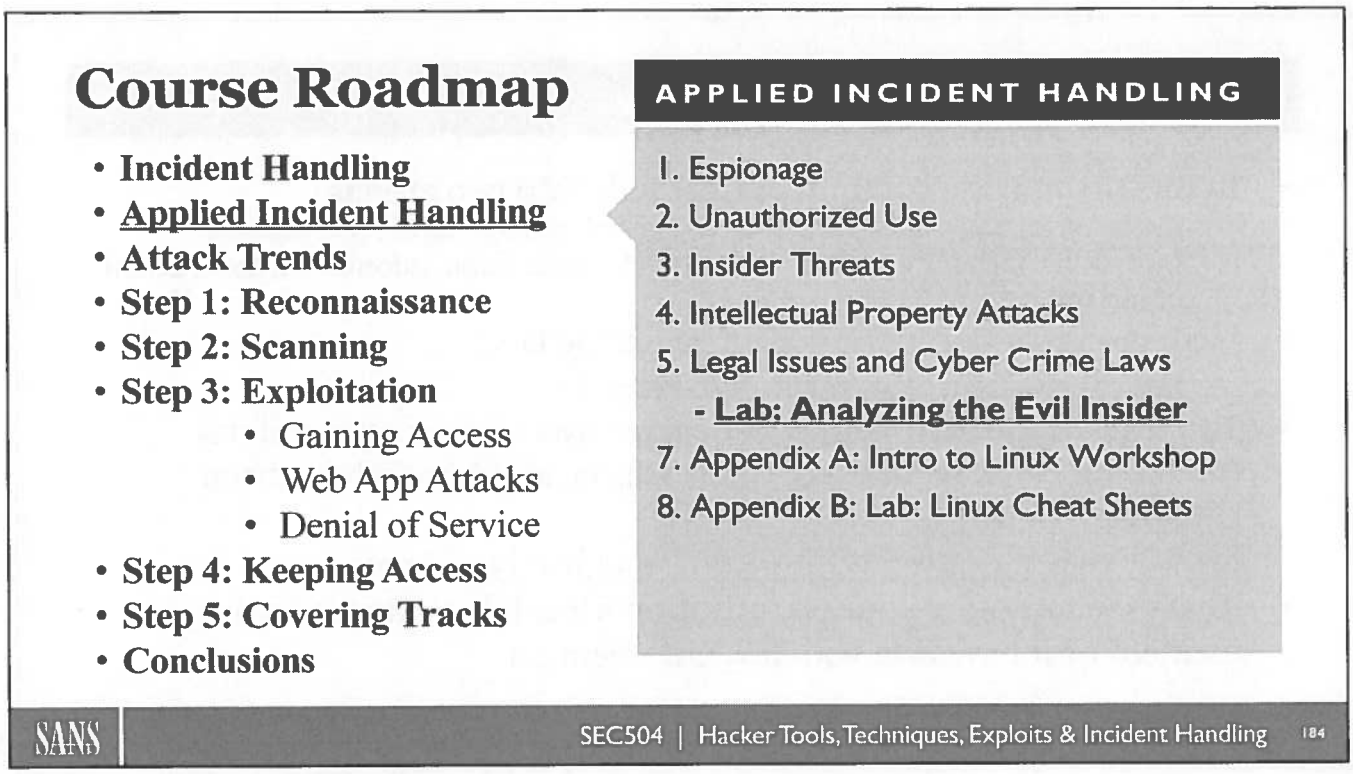
Many people who take this class are from other countries, whose laws we cannot cover due to time constraints. To address every country for each participant in this course, we'd take up an entire week just going over cyber crime laws. Thus, if your country is not covered in the list we'll address, and you have a significant interest in the cyber crimes laws of your country, feel free to contact your instructor off-line during a break, asking about the cyber crime situation in your country. Your instructor may have experience in that country or may be able to refer you to someone who does.

However, most U.S. laws can be accessed via http://www.justice.gov/criminal/cybercrime/.

Additionally, the Mosstingrett web site has the text of the cyber crime laws for over 40 different countries, at the URL included on the slide above. It includes each of the countries we'll discuss here, plus additional countries such as Norway, Mexico, India, China, and Israel.

Regardless of the country you are working in, always consult with a lawyer on sensitive or potentially sensitive issues.

© 2016 Ed Skoudis, John Strand



Next, to close out the day, let's do a lab that focuses on investigating an insider case.

© 2016 Ed Skoudis, John Strand

Lab: Analyzing the Evil Insider

- Now for a lab on detecting and analyzing an insider attack
- Some of the most destructive attacks in history have involved insiders
  - Robert Hanssen and Aldrich Ames are two examples
- Most organizations do not bother with detecting insiders
- However, one of the first steps many bad guys use is getting valid credentials
- Insider attackers often exploit a system > steal credentials > then use credentials against other systems
  - Telnet, FTP, Remote Desktop, SSH, and more
- Because of this, it is increasingly important that we develop ways to detect insider threats
- Analyze the evidence, answer the questions, and think of plans to detect insiders in your environment

One of the most difficult aspects of computer security is trying to detect insiders. If we look back in history, a large number of our most damaging attacks have been carried out by people with legitimate access to systems and sensitive data. Paradoxically, few organizations put any time or effort into detecting insider attacks. Why? Mainly because of two different reasons. Some organizations believe they can trust their people. This is a natural human tendency. We tend to think of people on our team, in our company, and within our organization, as "safe." It is a classic "us versus them" mentality. The other reason we tend to avoid detecting insiders is because it is difficult. In some situations, it is incredibly difficult.

There are two reasons we all need to start developing strategies for detecting insiders. The first is because a trusted insider can cause the most damage. The other reason is because it is going to become increasingly difficult to differentiate between an insider causing damage and an attacker who has valid credentials causing damage. This is so critical to understand, because one of the first things attackers try to do is gain access to systems using valid credentials. They will then be able to navigate through your environment as a valid user. No IDS/IPS/AV bypass will be necessary.

So, in this lab, we look at evidence of an insider attack. There are three key points of evidence; it will be up to you to review the data and answer some questions as you work your way through the lab scenario. The key of this lab, however, is to get you to start thinking differently about detection and prevention so that you can properly detect insider attacks, along with external attacks.

© 2016 Ed Skoudis, John Strand

Lab: Scenario and Evidence Files

- Monday morning, it comes to your attention that one of your co-workers, Joshua Winters, is in Utopia, Iceland, selling your Widget intellectual property to the competition
  - His user ID is jwinters
- Management wants the following questions answered:
  - Did he access any files related to ProjectX?
  - How did he get the data out of the network?
  - What other possibly sensitive files did he get?
- You have the following evidence files in the Windows\504_Insider_lab directory on your Class USB:
  - **504_File_Access.txt**: A dump of the file and folder access event logs
  - **504_Target_usb.txt**: The output of your drive monitoring script
  - **504_Target_Loggedin.txt**: The output of your script to log who is logged in to what systems
- This environment is well prepared because regular systems are monitoring for drives, users, and file/directory access auditing
- Analyze the evidence then answer the above questions

The scenario involves one of your users in Utopia, Iceland, trying to sell some of the secrets about your organization's new ProjectX widget. Oddly enough, one of your competitors brought this to your attention as she was contacted by Josh Winters with the offer of a sale.

Management, as can be expected, is in an agitated state. It has provided you with a list of questions, and you must answer them by close of business today. Analyze the evidence, follow through the steps of this lab, and write your answers here:

A) Did he access any files related to ProjectX?

B) How did he get the data out of the network?

C) What other possibly sensitive files did he get?

This environment is fortunate. They currently audit file and directory access record information about drives on the network, and capture who is logged into various systems.

All the commands used to create the evidence in this lab are on the following slides (where the evidence is analyzed) and were also provided near the top of each evidence file itself.

Finally, these commands are incredibly valuable in any incident. We use wmic extensively in this lab and in this class for the rest of the week.

© 2016 Ed Skoudis, John Strand

Lab: Checking Logins

- The power of WMIC and systems monitoring
- We can see that jwinters was logged onto the TEEVEE workstation
- This can help us tie together that he was logged in at the time the other evidence files were captured
- This can also be used to detect simultaneous logins
  - Useful to detect an attacker reusing compromised credentials
- The /node:@systems.txt syntax allows you to run this command against multiple systems
- Run this on your own system as administrator

```
C:\> wmic computersystem get username
```

```
###Created with wmic /node:@systems.txt computersystem get username /format:csv >
504_Target_Loggedin.txt###
Node,UserName
CHARLIE,TARGET\rhanssen
TEEVEE,TARGET\jwinters
VERUCA,TARGET\aames
META2008R2,TARGET\Administrator
```

Suspect account was logged on

One of the first things we need to review is whether Mr. Winters was logged in at the same time the other evidence files were collected. We can get some interesting information from the wmic command, pulling the computer system context from a variety of machines listed one per line in a systems.txt file:

```
c:\> wmic /node:@systems.txt computersystem get username /format:csv
```

The /node: allows us to run this command on a remote system. The @systems.txt allows us to run this command on every system listed in a file.

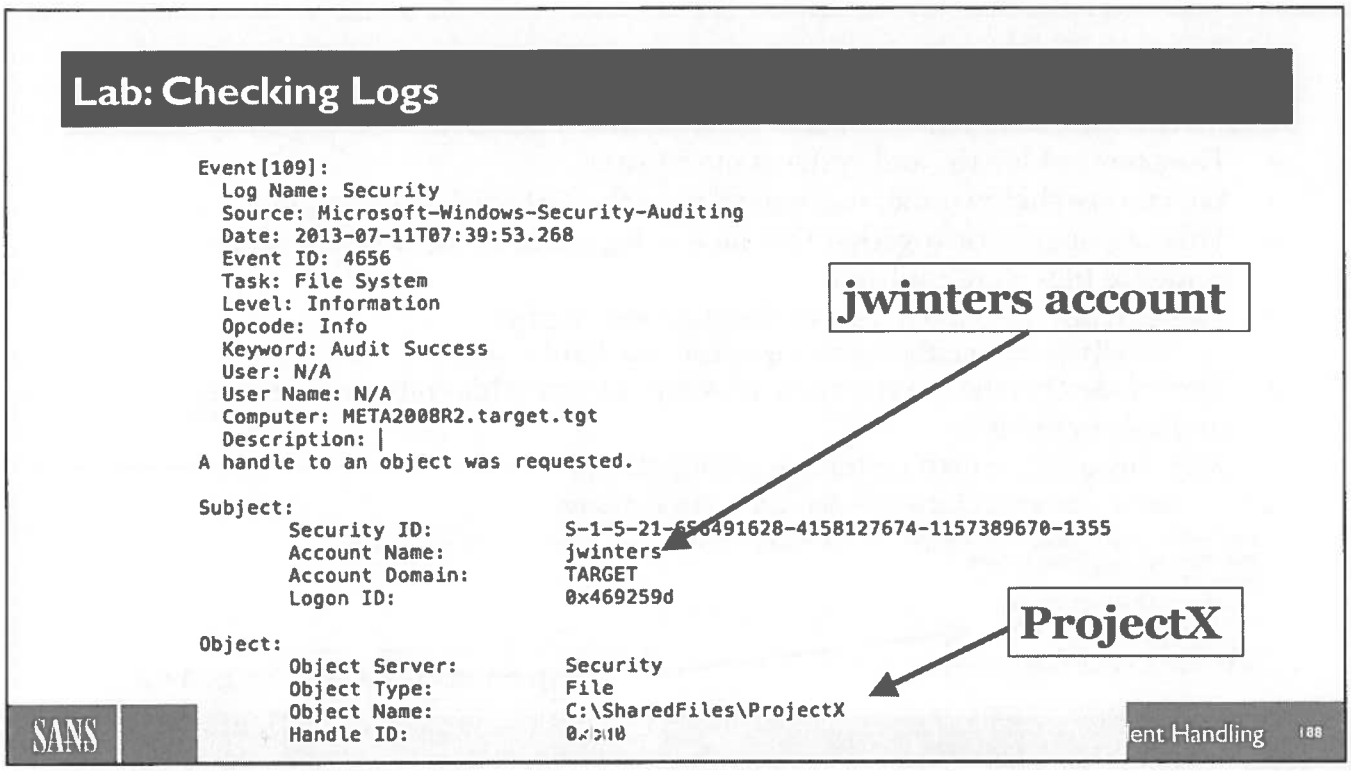
If you want, take a moment and run this command on your computer to get familiar with its output:

```
c:\> wmic computersystem get username
```

This command is so simple and can be critical to a variety of Windows-related incidents you handle. In this case, its output confirms that Mr. Winters was logged into a workstation called TEEVEE. However, there is more to this command. For example, it can be used to check for multiple simultaneous logins in your environment. Why would this matter? This would matter not only for detecting an insider, but also for detecting possibly compromised user accounts. As we said at the start of this lab, a growing trend used by attackers is using legitimate credentials to spread around a network. We have seen environments where the bad guys use the same credentials to log onto 30 or more systems via Remote Desktop to find systems with sensitive files, or find systems that are behind on patches or to find systems that are not part of the systems being monitored by an Internet Proxy.

Often, if there is an incident involving a Windows machine, this is one of the first commands we run.

© 2016 Ed Skoudis, John Strand



Next comes the most interesting part of this lab: reviewing the logs. Let's look at the 504_File_Access.txt file. The first question from management was did this user access anything related to ProjectX? A simple search in the log file confirms the insider, in fact, did gain access to the documents for ProjectX.

To see how we can verify this conclusion, open the file in an editor, such as Notepad, and do a search for ProjectX. Now, follow the progression of the following events: You will see the system check to see if the user has the correct permissions (Event ID 5145), then the account will request a handle to the object (Event ID 4656) then an access attempt to the file (Event ID 4663), then the handle to the object will be closed (Event ID 4658). This ties in with what we discussed earlier. Many times, an incident handler will have to tie together multiple logs entries to explain an event.

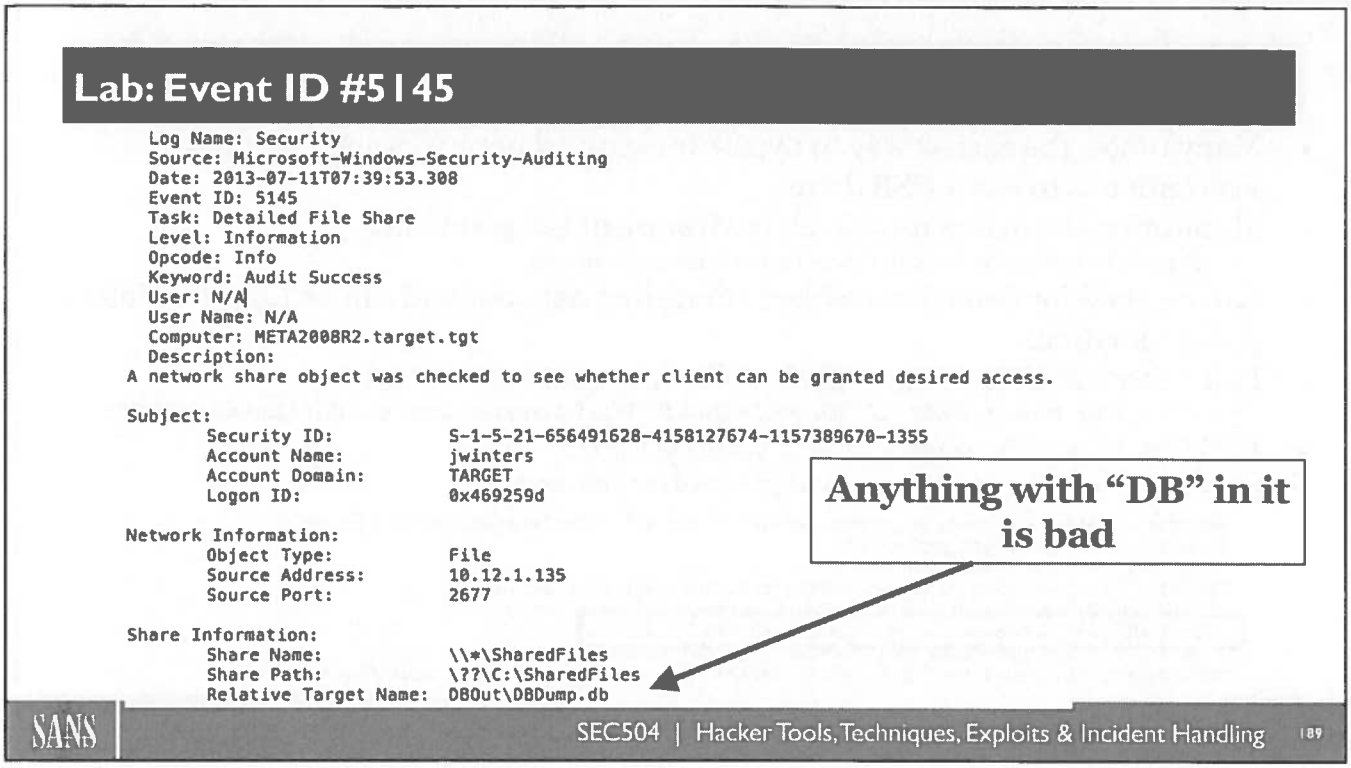
However, the next question takes more time. What other sensitive documents and directories did the attacker gain access to?

The key to all of this is the first Event ID of 5145.

Also, this evidence file was created with the following command:

```
c:\> wevtutil qe security /f:text > 504_File_Access.txt
```


© 2016 Ed Skoudis, John Strand



When event logging triggers based on accessing a file or a directory, a number of different events will be triggered. One will see the file resource request, another will track if the user has the appropriate permissions, but the one we want deals with successfully accessing files or directories.

The event ID number for that event is 5145. As we discovered in the previous slide, this is the first event ID that starts the logging for a user account requesting access to a file or directory. If we want a list of the files jwinters gained access to, we need to search on this ID and record the directories and files he accessed. Note only the files that appear to be interesting.

Simply open the log file with Notepad (or any other editor) and do a search for 5145.

A simple search for 5145 will pull all the Audit Success entries for accessing a file or a directory. Now, you should be able to identify all the sensitive files and directories the bad guy accessed.

The output for this section of the lab was created using the following command:

```
c:\> wevtutil qe security /f:text > 504_File_Access.txt
```

Note the above command was used on a Windows 7 system. If you are XP, the above command does not exist; you have to use the one below.

```
c:\> eventquery.vbs /L security
```

Lab: Checking Drives

- Many times, the easiest way to bypass air-gapped networks and Data Loss Prevention is to use a USB drive
- Monitoring the drives used in an environment is a good idea
  - Especially for highly sensitive portions of your environment
- Can be good for detecting insiders smuggling data out and can be useful to detect policy violations
- Pull historical USB usage with the following query command:
  - `c:\> reg query HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USBSTOR`
- Feel free to run the command on your system

```
C:\> wmic diskdrive get interfacetype,mediatype,model

##Created with wmic /node:@systems.txt diskdrive get interfacetype,mediatype,model
format:csv > 504_Target_usb.txt##
Node,InterfaceType,MediaType,Model
CHARLIE,IDE,Fixed hard disk media,VMware Virtual IDE Hard Drive
TEEVEE,IDE,Fixed hard disk media,VMware Virtual IDE Hard Drive
TEEVEE,USB,Removable Media,PNY USB 2.0 FD USB Drives
VEROCA,IDE,Fixed hard disk media,VMware Virtual IDE Hard Drive
META2008R2,SCSI,Fixed hard disk media,VMware, VMware Virtual S SCSI Disk Device
```


Finally, we want to check and see how the evidence left the environment.

Once again a very simple and useful wmic command will pull this information. Once again, the /node:@systems.txt allows us to run this command against multiple remote computer systems.

Note that a number of other devices, such as iPods, iPhones, and Android devices, register a new file system on the computer into which they are plugged.

If you want to try this on your own computer, run the following command without a USB drive in, then re-run it with a USB drive inserted into your USB port:

```
c:\> wmic diskdrive get interfacetype,mediatype,model
```

You can also find the historical USB usage by viewing the contents of the following registry key:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USBSTOR

But, this only works if a USB drive has been inserted.

For example, you can run the following on your system:

```
c:\> reg query HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USBSTOR
```


© 2016 Ed Skoudis, John Strand

Lab: Conclusions

- Did he access any files related to ProjectX?
  - Yes, he was able to access the entire folder because of weak permissions
- How did he get the data out of the network?
  - Via a USB drive
- What other possibly sensitive files did he get?
  - Database files, security files, stuff we don't want released, accounting's evil plans, recipes
- How can we do better?
  - Monitor things like logons, file access (sensitive files), drive usage with tools like Nagios, Tripwire, and SCCM in enterprises
- Key point: Simply reviewing IDS/IPS/AV is not enough!
- Detecting insider attacks and attackers with valid user credentials is essential to any security architecture

As we can clearly see from the evidence collected, Mr. Winters got access to a large amount of sensitive data.

It requires us to have a firm understanding of what is normal and being able to detect deviations from that norm. It also shows how the traditional security technologies we use today, such as IDS, IPS, and antivirus may not tell the entire story. It may require us to interact with the system-administrator team to review systems-monitoring data. It may require us to set up tools and techniques to more effectively detect threats, like the trusted insider.

© 2016 Ed Skoudis, John Strand

504.1 Conclusions

- We completed the policy, procedure, and analytic fundamentals
  - These are crucial underpinnings for successful incident handling
- Tomorrow, we focus on the technical attacks and defenses
  - Step-by-step
  - Offense must inform defense

So, there you have it: the incident-handling process. Keep in mind that policy and procedure are absolutely essential in security, especially incident handling. Tomorrow, we focus on how computer attackers undermine our systems and how to detect them and defend at each stage of the attack.

© 2016 Ed Skoudis, John Strand

Course Roadmap

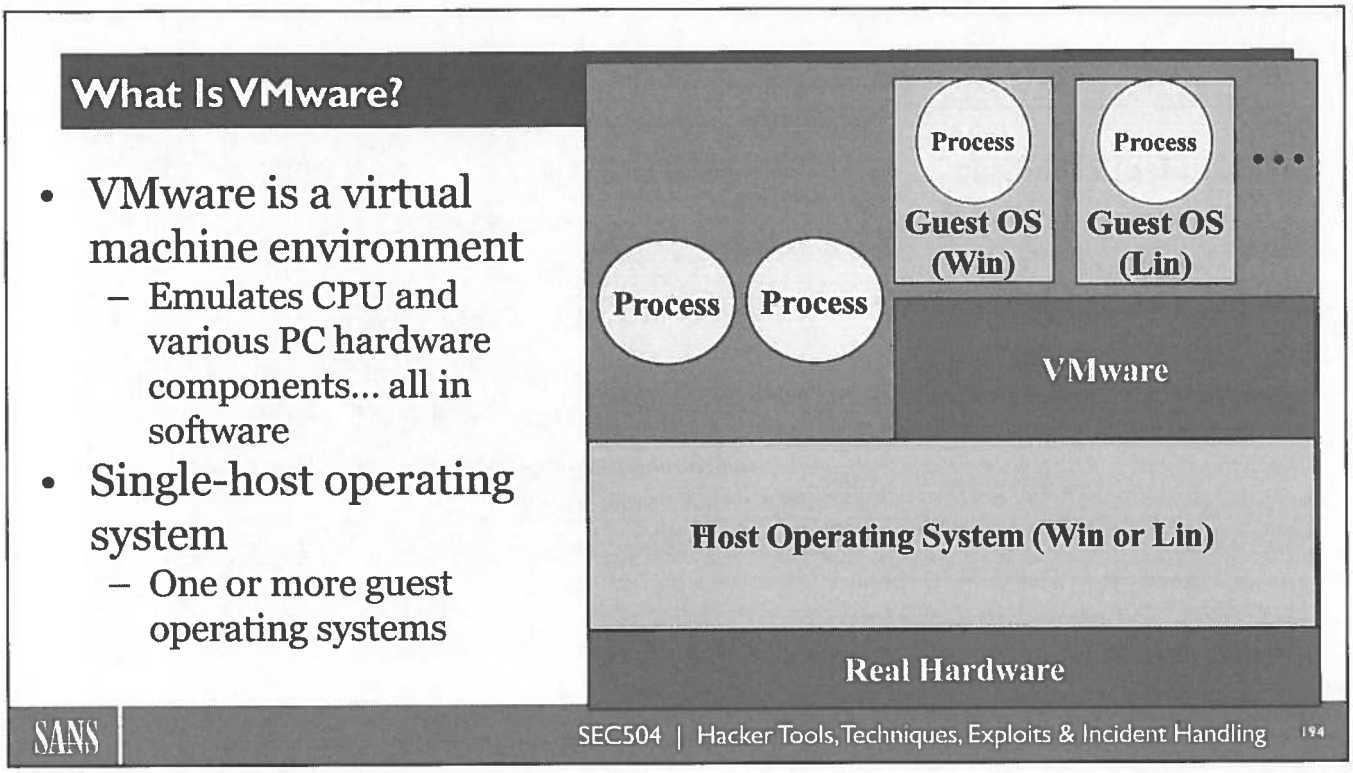
- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

APPLIED INCIDENT HANDLING

1. Espionage
2. Unauthorized Use
3. Insider Threats
4. Intellectual Property Attacks
5. Legal Issues and Cyber Crime Laws
   - Lab: Analyzing the Evil Insider
7. **Appendix A: Intro to Linux Workshop**
8. Appendix B: Lab: Linux Cheat Sheets

Next, we are introduced to VMware Workstation and Player, which are virtual machine products from VMware.

© 2016 Ed Skoudis, John Strand



VMware is simply an emulator for a PC, all created in software. You install the VMware program on top of another operating system, known as the host operating system, such as Windows or Linux. Then, you can boot up virtual computers within VMware, each of which is referred to as a guest operating system or a virtual machine. Each guest has its own memory allocation, virtualized network adapters, hard drive(s), and other hardware components. The different guests and the host appear to be truly independent operating systems, all running on the same hardware.

In my own incident-handling and malware analysis activities, I rely extensively on VMware. I use it to practice hacking between virtual systems, perform forensics analysis of compromised machines, and safely test malware specimens. In fact, I don't know how I could do my job today without VMware or another similar virtual PC product.

© 2016 Ed Skoudis, John Strand

VMware Uses

- Virtual machines are inherently useful for
  - Incident response
  - Malware analysis
  - Digital forensics
  - Ethical hacking
  - Practice hacking
- This class uses VMware Workstation, VMware Player, or VMware Fusion
  - Other virtual machine platforms aren't officially supported, but you are free to try them

You can use VMware for so many different applications, including incident response, malware analysis, digital forensics, ethical hacking, and even practice hacking. VMware can be applied to many in-depth information security tasks.

© 2016 Ed Skoudis, John Strand

VMware Machines

- VMware machines consist of files in the host operating system, typically grouped into a single directory for each virtual machine
  - .vmx = Virtual machine's configuration
  - nvram = Stores the state of the virtual machines BIOS
  - .vmdk = Stores the virtual disk file, the hard drive image(s) of the virtual machine
  - .vmss = Suspended state file, for a paused virtual machine
  - .vmsn = Snapshot file, used for taking a snapshot of the system state for restoring it later

So, what is a virtual machine inside of the VMware application? It consists of a directory, with a bunch of files holding information about that guest operating system installation. The virtual files making up a VMware guest include files with these suffixes:

.vmx = Holds the virtual machine's configuration, including hardware and network settings.

nvram = Stores the state of the virtual machine's BIOS, including the BIOS boot program, clock settings, and so on.

.vmdk = Stores the virtual disk file, that is, the hard drive image(s) of the virtual machine. A single virtual machine may have multiple virtual drives, so there could be more than one .vmdk file.

.vmss = Holds the suspended state file for a paused virtual machine. This suspended state includes a copy of RAM and the current console screen view.

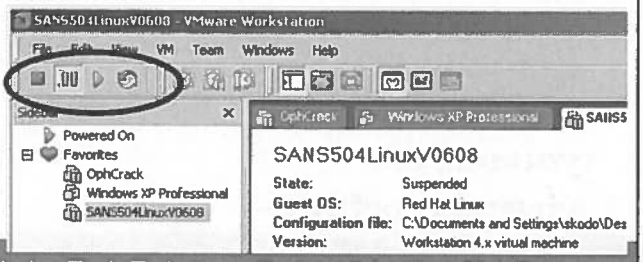
.vmsn = Stores a snapshot file, used for... well... taking a snapshot of the system state for restoring it later. This snapshot feature is immensely useful, especially when analyzing malware. Before I run malware that could be destructive, I take a snapshot. Then, if the malware hoses the machine entirely, I can roll back to the most recent snapshot, restoring the system easily without having to rebuild!

You can back up the entire contents of a virtual machine, RAM, hard drive, and all, by simply creating a directory with a copy of all of these files. You can even zip it up for safe keeping.

© 2016 Ed Skoudis, John Strand

Booting and Controlling Virtual Machines

- For VMware Player, invoke guest machine by simply opening it
  - When you close the Player, it suspends the guest
  - When you open that guest again, it resumes where you left off
- For VMware Workstation, using the VCR-like controls, you can
  - Stop a virtual machine
  - Suspend (pause) a virtual machine
  - Boot or resume a virtual machine
  - Reset a virtual machine (reboot)
  - Take a snapshot of a virtual machine
  - Revert to a snapshot



VMware Workstation includes VCR-like controls for running guest machines. VMware Player does not. With the Player, you can simply invoke a virtual machine by opening it. When you close the Player, it essentially suspends the guest machine. Upon re-opening, the guest will be activated in the state where you last left off.

In VMware Workstation, each virtual machine can be easily controlled from within VMware using the VCR-like controls. You remember VCRs, right? You can stop a virtual machine by hitting the red Stop button. Hitting Stop, however, without gracefully shutting down the system is the equivalent of pulling the power cord on a physical system. Be careful! Your drive could get out of synch, and other problems could occur if you just hit Stop. Instead, you might want to suspend the virtual machine using the Pause button, which is akin to hibernating it. That operation is quite safe.

Also, you can boot a virtual machine by hitting the green play button.

You can even reboot a virtual machine by hitting the red and green arrow button. Again, this is like applying a hardware reset on a real system, so be careful.

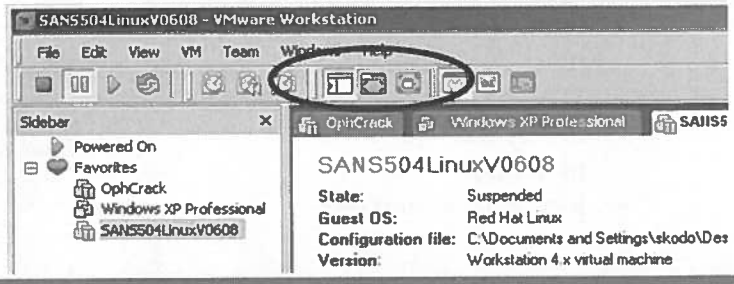
Finally, you can snapshot the virtual system or revert it to a previous snapshot using the Snapshot and Revert buttons.

© 2016 Ed Skoudis, John Strand

Controlling Screen Views

- VMware Workstation supports three different screen view modes
  - **Navigation Bar Mode**: Shows favorite virtual machines
  - **Full Screen Mode**: Virtual machine takes up entire screen
  - **Quick Switch Mode**: Provides tabs to switch between machines

Remember: CTRL+ALT gets you out of a virtual machine



VMware Player does not offer many options for screen views. The guest is a window on top of the host operating system. When running VMware Workstation, the guest can interact with its user on the desktop using three different views:

- **Navigation Bar Mode**: Shows your favorite virtual machines defined on the system. These are essentially shortcuts to the guests that you use the most. This view is the easiest for jumping between different virtual systems running simultaneously.
- **Full Screen Mode**: In this mode, the virtual machine takes up the entire screen. This mode makes it look like (from a user interface perspective) the virtual machine is the only thing running on the box. It hides the host operating system from the GUI entirely.
- **Quick Switch Mode**: Provides tabs to switch between machines at the top of the screen, while hiding the host operating system GUI. It's like a blend of the other two modes.

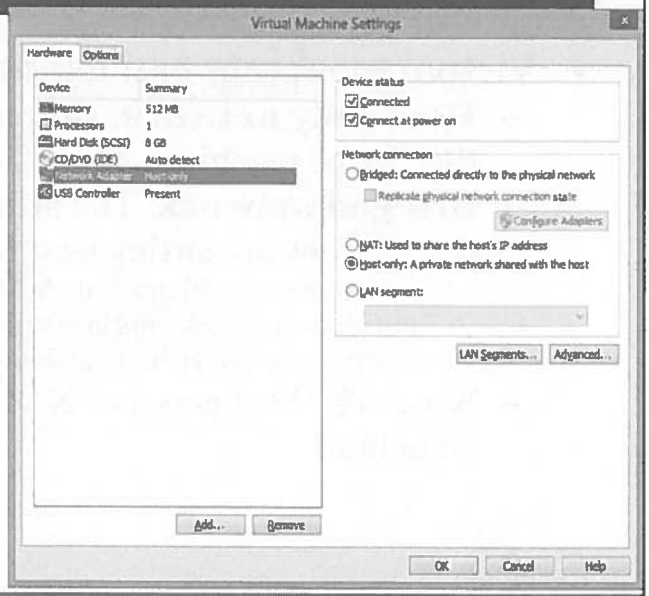
Always remember that you can hit CTRL+ALT to jump out of virtual machines back into the host, regardless of the screen mode you are using.

Also, to send a virtual machine a CTRL+ALT+DEL, you need to go to the VMware window and select VM→Send Ctrl+Alt+Del.

© 2016 Ed Skoudis, John Strand

VMware Configuration Options

- In VMware Workstation or Player, under VM→Manage→ Virtual Machine Settings
  - Or hit CTRL+D
- Most useful controls:
  - CD/USB
    - Can change it to physical optical disk or mount ISO image
  - Network adapter
    - See next slide



In VMware Workstation or Player, to adjust any of the virtual-hardware settings of a guest operating system, go to VM→Manage. There, you can adjust the amount of memory devoted to your virtual system. Also, you can mount a physical CD-ROM, or even an ISO image of a CD from your file system. That's particularly helpful.

One of the most important settings in these configuration options is associated with the network adapter: your Ethernet configuration.

© 2016 Ed Skoudis, John Strand

VMware Network Options

- Virtual machines can use one of three network options:
  - **Host-only network**: Nothing other than the host OS can get to the virtual machine across the network
  - **Bridged network**: The host and virtual machines behave as though they are sitting next to each other on a switch
    - Introduces virtual machine MAC addresses on the LAN
    - Puts host network interface in promiscuous mode (to capture traffic destined for the virtual machines)
  - **NAT**: The host acts as a NAT device, which the virtual machines sit behind

Virtual machines can use one of three network options, as follows:

- **Host-only network**: With this option, nothing other than the host OS can communicate with the Virtual machine.
- **Bridged network**: With this configuration, the host and virtual machines behave as though they are sitting next to each other on a switch. This introduces the virtual machine MAC address on the LAN. Also, it puts the host network interface in promiscuous mode (to capture traffic destined for the virtual machines, the host will have to grab packets destined for MAC addresses that don't match the hardware address). Keep this in mind! I've had many students freak out when their network interface is in promiscuous mode, thinking an attacker installed a sniffer. However, in reality, these students put their interface into promiscuous mode themselves by selecting bridged networking. It's not really a security risk. Also, whenever I'm hacking from a virtual machine across a real network, I always use this mode. I don't want any network address translation to get in the way of my packet-generation tools.
- **NAT**: In this mode, the host acts as a NAT device, which the virtual machines sit behind. All packets get their source IP translated so they appear to have come from the host instead of the guest operating system.

© 2016 Ed Skoudis, John Strand

VMware Networking – Watch Out!

- In your host OS, don't alter virtual adapters (VMneto, VMnet1, and VMnet8) unless you know what you are doing!
  - Don't configure them for DHCP or hard-coded IP addresses
    - The only exception is for a Windows host to communicate with VMnet1, as described next
- Besides that, all virtual adapter configuration should be done in the virtual machine!

When you install VMware, your host operating system will have three new network adapters created: VMnet0, which is used for bridged networking; VMnet1, which is used for host-only networking; and VMnet8, used for NAT.

Be careful with these! Changing their defaults could seriously impact your networking in the host and in the guest systems. I advise you to avoid setting them for DHCP or even hard-coded addresses unless you really know what you are doing. The only thing I recommend is that you alter VMNet1 if you want to communicate between guest and host in host-only mode when you don't have physical access to a switch. You'll see why on the next slide.

© 2016 Ed Skoudis, John Strand

Communicating with a Windows Host

- Sadly, Windows disables real interfaces if there's no link
- So, under default configs, the host and guest cannot communicate, because the host disables its own interface
- Thus, to communicate between guest OSs and a Windows host when there is no physical link, complete four steps:
  - One in the Host OS (if you are using VMware Player 3.0 or later or VMware Workstation 7.0 or later, skip this step)
  - Two inside of the VMware program (not in the guest OS)
  - One in the guest OS

Sadly, in a Windows host operating system, the physical network interface is disabled if you don't have a connection to a switch or hub. The OS auto-senses that there's no network and disables the networking if it's not physically there. That auto-disabling of the media messes up network communication between the host and guest operating system, unless we configure VMnet1 to allow such connections for host-only networking.

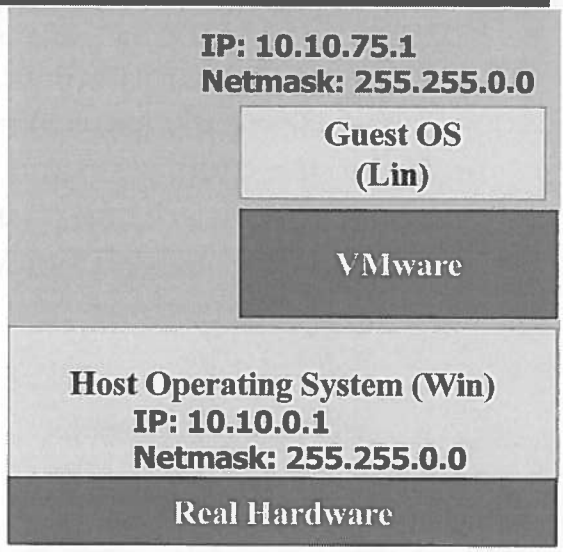
To accomplish this, we apply a four-step process:

- One step in the host OS (you can skip this step if you are using VMware Player 3.0 or later or VMware Workstation 7.0 or later)
- Two steps inside of the VMware program (not in the guest OS)
- One final step in the guest OS

© 2016 Ed Skoudis, John Strand

Here's the Scenario

- We want to set up our network so that it has the following characteristics:
  - No network link (physically unplugged)
  - Host and guest want to communicate with each other
  - Host OS (Windows) has an IP address of 10.10.0.1, netmask 255.255.0.0
  - Guest OS has an IP address of 10.10.75.1, netmask of 255.255.0.0



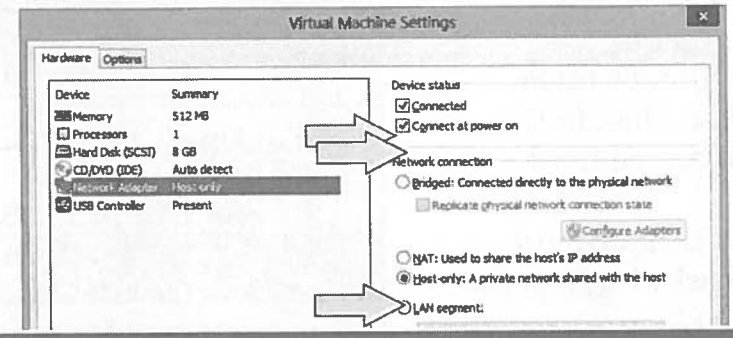
Here's our scenario: We assume that the host operating system is Windows, which we give an IP address of 10.10.0.1 and a netmask of 255.255.0.0.

We want it to communicate with a Linux guest OS with an IP address of 10.10.75.1, netmask of 255.255.0.0, even when there is no physical network interface connection. To do this, we have to configure VMnet1 in the host operating system to allow the communication. Let's look at our four step-process.

© 2016 Ed Skoudis, John Strand

Step 1 – Setting Up VMware for Windows Host-Only Comm

- Prepare VMware settings
  - Bring up VMware settings by hitting CTRL-D
  - Select Host-only networking
  - Make sure Connected and Connected at Power On are selected
  - If any options are grayed out, boot guest machine, make changes



IF YOU ARE USING A VMWARE PLAYER VERSION LESS THAN 3.0 , GO TO THE NEXT SLIDE.

For step 2, you need to run VMware. Inside VMware, go to VM→Settings, and click Network Adapter. Now, set it to the Host-only mode.

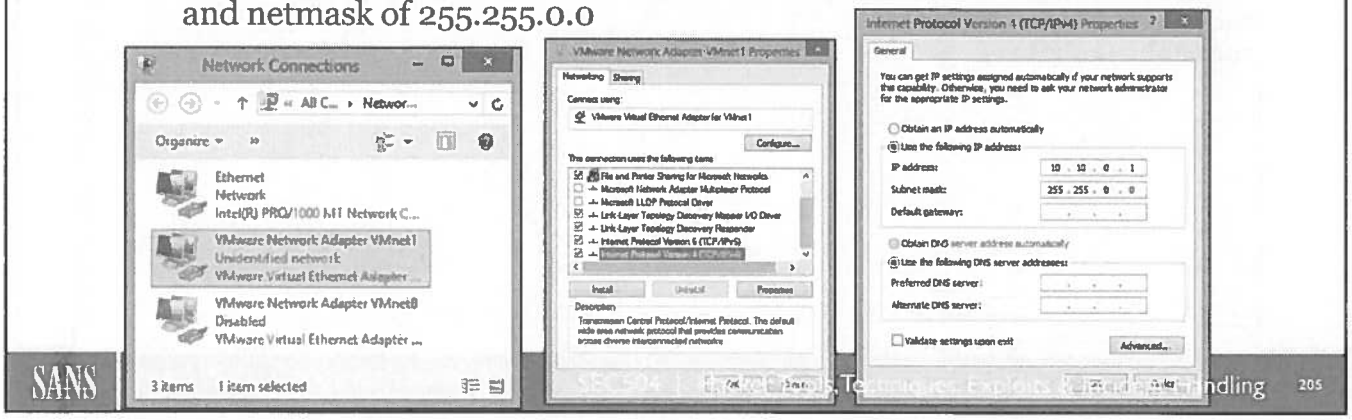
Also, make sure Connected and Connected at Power On are selected.

If they are grayed out, start up your guest machine, and then make these changes.

© 2016 Ed Skoudis, John Strand

Step 2 – Setting Up Windows Host for Host-Only Communication

- Set VMNet1 in host OS (Windows)
  - In Windows host OS, run this command (as admin) to bring up network adapters:
  - `C:\> ncpa.cpl`
  - Choose VMnet1, Select Internet Protocol, set the IP address to 10.10.0.1 and netmask of 255.255.0.0



For step 3, we need to set up VMnet1 in the host operating system. In your Windows host machine, at an admin-level command prompt, bring up a screen showing your network adapters by running

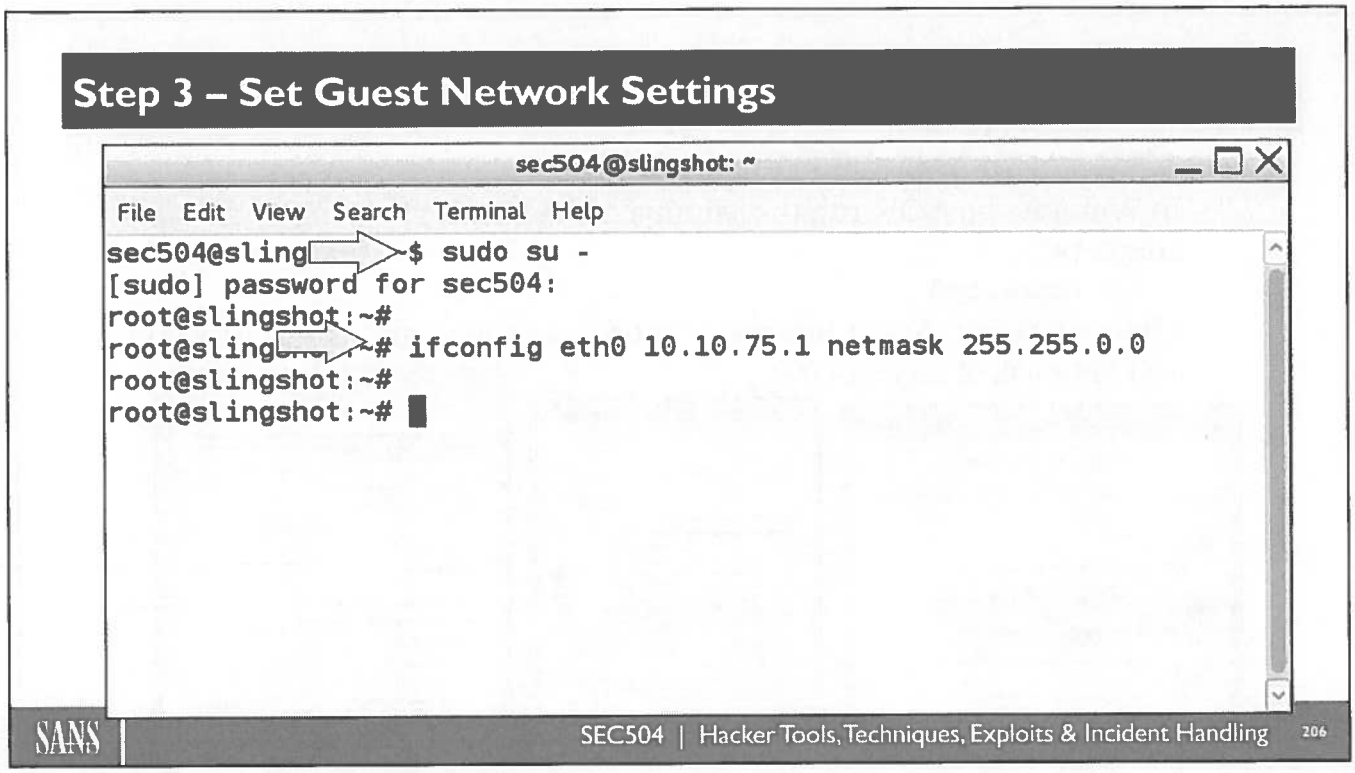
```
C:\> ncpa.cpl
```

Now, select the VMnet1 interface, click Properties, choose Internet Protocol Version 4 (TCP/IP), and click Properties.

Set the IP address and netmask as desired (10.10.0.1 and 255.255.0.0, respectively).

VMware may have automatically given you this IP address. If that is the case, verify that you have the address 10.10.0.1 associated with VMnet1.

© 2016 Ed Skoudis, John Strand



Now, we configure the Linux VM so it can communicate with the host for our labs.

First, we need to become root:

```
$ sudo su - (Then enter "sec504" as the password)
```

```
# ifconfig eth0 10.10.75.1 netmask 255.255.0.0
```

This sets your address, but does not keep it across a reboot of the virtual machine. Still, for temporary use, it's a great workaround for the VMware bug.

© 2016 Ed Skoudis, John Strand

Now Ping

- You should be able to ping back and forth between the host and guest across VMNet1
  - In host:
    `C:\> ping 10.10.75.1`
  - In guest:
    `$ ping 10.10.0.1`
- A personal firewall may block this access
  - Disable it if you have one
  - To disable the built-in Windows firewall, run:
    - C:\> netsh firewall set opmode disable
    - C:\> netsh advfirewall set allprofiles state off (For Windows 8+ systems)
  - To disable iptables in the course VMware image, run:
    - # iptables -F
  - Or at least allow access between guest and host OSs

Finally, we should be able to ping. Run ping from the host operating system to ping the guest:

```
C:\> ping 10.10.75.1
```

It should work.

Also, you can ping the host from the guest:

```
$ ping 10.10.0.1
```

If it's not working, you may have a personal firewall blocking the packets in either the host or guest operating system. Disable the personal firewall, and try pinging again.

To disable the built-in Windows firewall, you can run:

```
C:\> netsh firewall set opmode disable
C:\> netsh advfirewall set allprofiles state off (For Windows
   8+ systems)
```

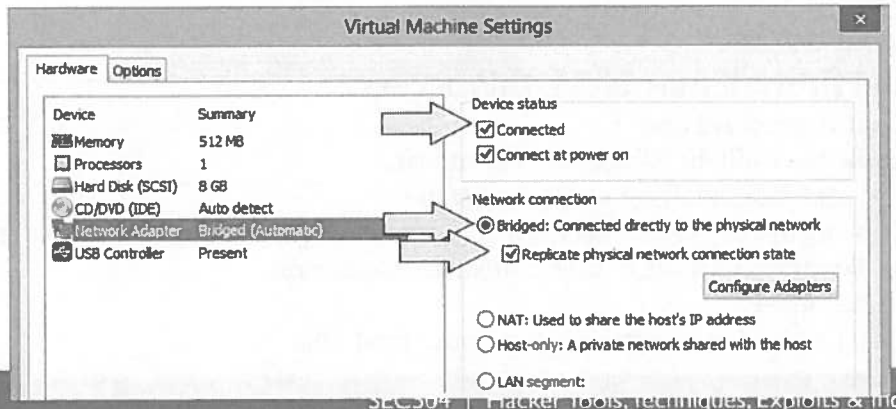
To disable iptables in the course VMware image, you can run:

```
# iptables -F
```

© 2016 Ed Skoudis, John Strand

When Going Back to Physical Connections... DON'T FORGET

- When you go back to a physical connection to a switch or hub, set your VMware networking to Bridged mode
- Select Bridged instead of Host-Only, and click OK



Remember, if you ever want your virtual machine to communicate with the outside world across a physical network, you must go back into the network settings and select Bridged mode.

© 2016 Ed Skoudis, John Strand

VMware Conclusions

- VMware is very powerful!
- Be careful with those network settings!
- Enjoy!

That's VMware. It will serve you well in this class, and I hope you find it useful on the job.

That concludes this introduction to VMware. You can either leave now or stay for the Intro to Linux mini-workshop.

© 2016 Ed Skoudis, John Strand

Intro to Linux for Hacker's Workshop

- Linux is powerful but is also complex
- Still, even with little exposure to Linux, you can fully participate in the hacker tools workshop
- This course segment is designed to get you up to speed with Linux
- After this, you won't be an expert, but you'll be ready to go for the workshop
  - Our focus here is on practicality, not theory

To fully participate in this class, you need a basic working knowledge of Linux. We're not expecting you to be an expert, by any means. Everything you need to know about Linux for the workshop will be covered in this introductory workshop.

We will not be covering Linux installation. You should have done that before coming to the session, as described in the class requirements.

© 2016 Ed Skoudis, John Strand

Fun Ease-of-Use Shell Tips

- The default shell of many Linux distros is bash, which has many ease of use features, including:
  - Command history, accessible via up and down arrows
    - Then use left and right arrows to position cursor to edit command
  - Tab auto complete for directory and filenames
    - Tab once to expand to unique
    - Tab twice to show non-unique matches
  - CTRL-R history search
    - Press CTRL-R and then type characters to find recent commands with those keystrokes in that order
  - CTRL-L to clear screen
  - CTRL-C to abandon current command (no need to press Delete key)
  - Home key to go to start of command line, End key to go to end, useful for editing long commands

Throughout this session, we use bash as a command shell, one of the most common command shells in Linux distributions today. This shell includes many ease of use features that make interacting with Linux simpler. You should memorize each of these items, as they will save you much time and effort, making Linux a lot friendlier for you.

Bash, like many other shells, remembers your shell history, letting you access it by pressing the up and down arrows to access and edit recent commands, which you can rerun by simply pressing Enter.

After you choose a previous command, you can press the left and right arrow keys to position your cursor to edit the command.

Also, bash supports tab auto-complete for the names of directories and files. When accessing something in the file system, just press Tab for the shell to expand it to a unique name that matches what you've typed so far. If there are multiple items that match what you've typed (that is, there is nothing unique yet), you can press Tab again to show the names of all files or directories in your current working directory that match what you've typed so far. That is, Tab expands to a unique value, and Tab-Tab shows all items that match what you've typed so far if nothing is unique.

You can also search your history in bash by pressing CTRL-R at the start of a command line. Then, start typing characters, and bash jumps back to the most recent command that has the characters you typed in that order. You can then press Enter to rerun that command or the left or right arrow keys to edit the command.

The CTRL-L option clears the screen, or you can simply type **clear**. The CTRL-C command lets you abandon the current command and get back to the command prompt. There is no need to delete the current command by holding down the backspace or Delete keys. Just press CTRL-C to get rid of the current command.

The Home key included on some keyboards lets you jump to the beginning of a command line, whereas the End key lets you jump to the end. These options can help you jump around in long commands to make altering them easier.

© 2016 Ed Skoudis, John Strand

Intro to Linux Topics

- Account Stuff (Logging In, useradd, passwd, su, whoami, terminal control)
- File System Stuff (structure, cd, pwd, ls, abs and rel referencing, mount, eject, mkdir, cp, find, locate, gedit, cat, less)
- Running Programs (PATH, which, ./, ps, jobs)
- Network Stuff (ifcfg-etho, restarting interfaces, ifconfig, ping, netstat)
- Building Tools (tar, configure, make)
- Other odds-and-ends (grep, man, info, shutdown)

Here's an outline describing the topics we'll cover. We'll start with Account stuff.

© 2016 Ed Skoudis, John Strand

Logging In as Root Versus Non-root (useradd)

- For almost all activities, you should log in as a non-root user
  - Create a user by using the useradd command:
    `# useradd -d [home_dir] [login]`
  - A "#" prompt means you are root
  - A "$" or other prompt means you aren't
- User's home directory is where that user is placed after logging in
  - The home dir also stores that user's files

Go ahead and create a non-root account on your system.

Keep an eye on your prompt. If it's a "$", you just aren't root. If it's a "#", you are root.

As root, type the following:

```
# useradd -d /home/fred fred
```

The login account "fred" will be created, with a home directory of /home/fred. The system will automatically assign a non-root userID to the account. The userID is just a number associated with this account for the purposes of assigning permissions. The home directory is where config files and other personal files for this account are stored.

© 2016 Ed Skoudis, John Strand

Changing Passwords (passwd)

- The passwd command is used to change passwords
- Any user can type "passwd" to change his or her own password
  - The user is prompted for the new password twice

```
$ passwd
```

- Or to change any user's password, root can type:

```
# passwd [login_name]
```


Currently, the fred account we created cannot be used because we haven't yet set a password. (The password isn't blank; the account is just disabled until we enter a password.) We need to set a password for the new fred account by typing:

```
# passwd fred
```

[type account password here]

[retype account password to verify]

If fred wanted to change his own password, fred would type (from the fred account):

```
$ passwd
```


© 2016 Ed Skoudis, John Strand

Changing Accounts (su and whoami)

- Do everything as a non-root user, except for things you truly need root for
  - For most of the tools used in this class, you'll need root privs
  - If you do need root, use the sudo command
  - To get a root prompt, run:

```
$ sudo su -
[type your password]
```

  - If no account_name is given, root is assumed
- The command whoami shows which account you are using

```
$ whoami
```

- For more details, use the id command
- On many Linuxes, UID 0 accounts cannot ssh in directly
  - Ssh in as another user, and then su your way to UID 0

If you are logged in already, you run commands with the privileges of another account via the sudo command.

To get a root prompt, you could run:

```
$ sudo su -
```

Then, you can type in your accounts password, and it if has sudo rights to run a shell as root, you'll get a root prompt on your system.
The whoami command shows who you are currently logged in as.

Type the following:

```
# whoami
```

Given the # prompt at the beginning of this command, you will likely see "root" on the output. Try:

```
# su fred
```

(Notice that the prompt changed!)

```
$ whoami
```

Here, you should see that you are now "fred". You can exit your most recent su by running:

```
$ exit
```

(The exit means that we are leaving the user fred, and returning to root.)

```
# whoami
```

Now, you should be root again (note the # prompt).
For even more details about your current user id and privileges, use the id command:

```
# id
```

© 2016 Ed Skoudis, John Strand

Intro to Linux Topics

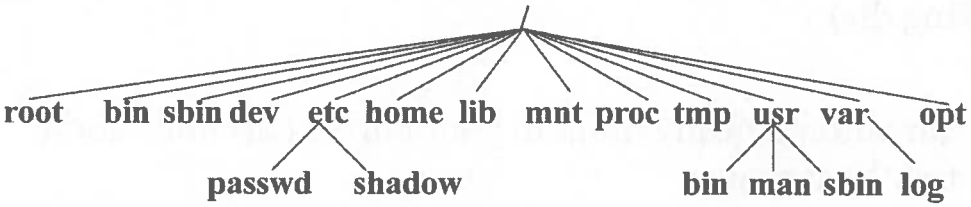
- Account Stuff (Logging In, useradd, passwd, su, whoami, terminal control)
- File System Stuff (structure, cd, pwd, ls, abs and rel referencing, mount, eject, mkdir, cp, find, locate, gedit, cat, less)
- Running Programs (PATH, which, ./, ps, jobs)
- Network Stuff (ifcfg-eth0, restarting interfaces, ifconfig, ping, netstat)
- Building Tools (tar, configure, make)
- Other odds-and-ends (grep, man, info, shutdown)

Here's our pesky outline again. Let's cover File System Stuff next. This is the longest section, simply because so much of Linux is oriented around its file system.

© 2016 Ed Skoudis, John Strand

Linux File System Structure

- The top of the file system is called /
- A bunch of things are under slash
- Here is a representative sample of what's under /
  - Varies for different versions of Linux



- Executable program are stored in /bin and /sbin.
- /root is the root login account's home directory. This is hugely important because if you log in directly as root, this will be your initial location in the directory structure. If you log in as an individual user other than root, you'll be put in that user's directory, typically somewhere inside of /home.
- /dev stores devices (drives, terminals, etc.)
- /etc holds configuration items, like the account information (stored in /etc/passwd) and hashed passwords (stored in /etc/shadow).
- /home contains user's home directories.
- /lib contains common libraries.
- /mnt is where various remote and temporary file systems (CD-ROMs, floppies, etc.) are attached.
- /proc is a virtual file system used to store kernel info.
- /tmp is for temporary data, and is usually cleared at reboot.
- /usr holds user programs and other data.
- /var hold many different items, including logs (/var/log/).
- /opt stores optional items and is often a location for specialized tools that have been added to a distribution.

© 2016 Ed Skoudis, John Strand

Navigating the File System (cd and pwd)

- You can move around the file system using the cd command

```
$ cd [directory-name]
```

- Parent directory (up one level) is called ".."

```
$ cd ..
```

- To see where you are, use the "pwd" command (short for print working dir)

```
$ pwd
```

- You can automagically jump to your current account home directory by typing:

```
$ cd ~    (or just "cd" by itself)
```

SANS | SEC504 | Hacker Tools, Techniques, Exploits & Incident Handling 218

If you are following along, let's change to the top-level directory:

```
$ cd /
$ pwd
```

What do you see?

```
$ cd ~
$ pwd
```

What do you see?

© 2016 Ed Skoudis, John Strand

Looking at Directory Contents (ls)

- The "ls" command shows directory or file details
  - By itself, shows regular files
  - With the "-a" flag, ls shows all files (including those that start with a ".")
  - With the "-l" flag, ls shows details (permissions, links, and so on)

```
$ ls -la
```


If you are following along, type:

```
$ cd /etc
$ ls
```

What do you see?

```
$ ls -la
```

You now are looking at details associated with your /etc directory. System configuration information is stored here.

© 2016 Ed Skoudis, John Strand

Absolute and Relative Referencing of Files

- You can refer to files with their full path in the file system (absolute referencing; everything starts with "/")

```
$ cd /etc/init
$ pwd
```

- Or you can refer to files relative to your current working directory (everything starts assuming where you are currently located)

```
$ cd /etc
$ cd init
$ pwd
```


For any file, you can refer to it using the relative reference (based on your current working directory), or the absolute reference.

Try the following, using absolute referencing for the directory:

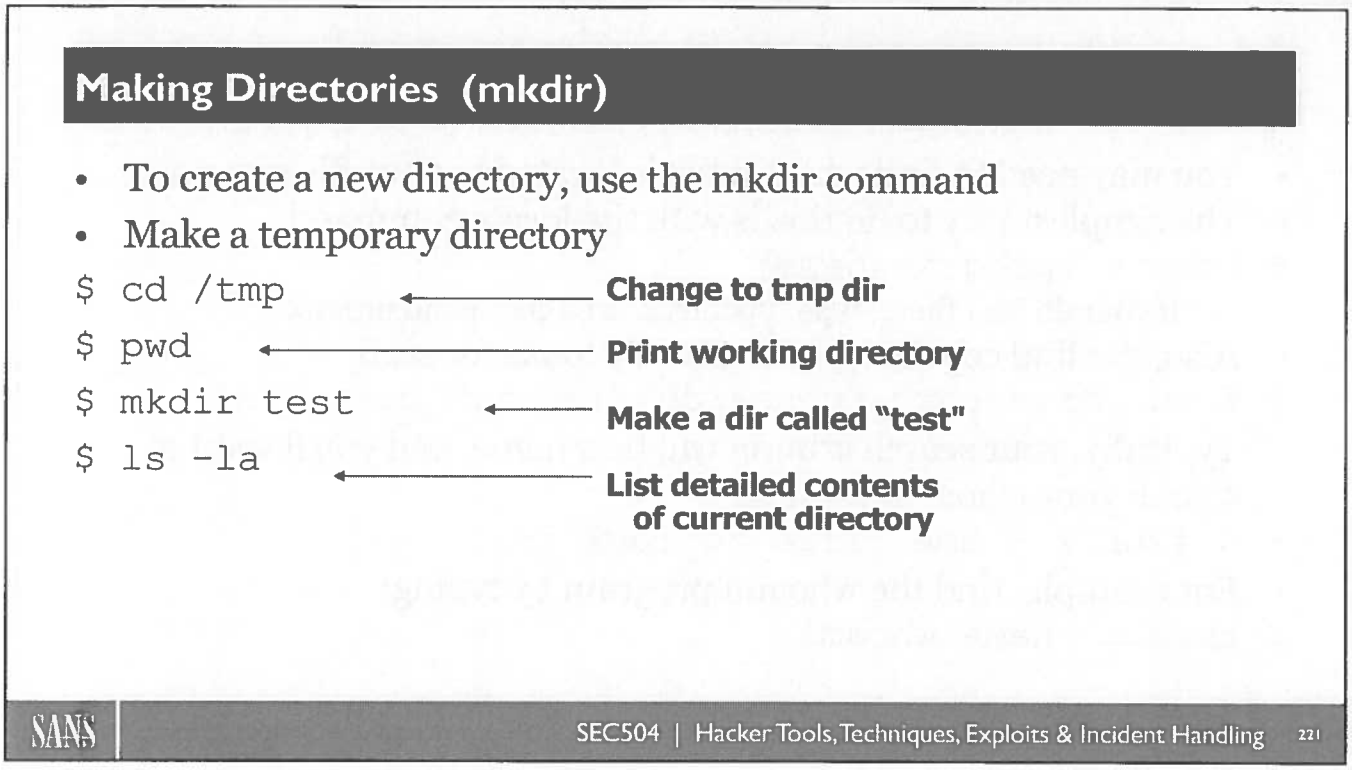
```
$ cd /etc/init
$ pwd
```

Or you can do it in two steps, using relative references:

```
$ cd /etc
$ pwd
$ cd init    ← Note that we dropped the leading /
$ pwd
```

What do you see?

© 2016 Ed Skoudis, John Strand



To create a directory, use the mkdir command. Let's create a test directory in our /tmp directory.

```
$ cd /tmp
$ pwd
$ mkdir test
$ ls -la
```

What do you see?

© 2016 Ed Skoudis, John Strand

Finding Files (locate and find)

- You may need to find where a file is located in your file system
- The simplest way to do this is with the locate command

```
$ locate [program_name]
```

  - If your db isn't there, type "updatedb" at a command prompt

- Also, the find command exhaustively looks for stuff

```
$ find [directory_to_search] [search_criteria]
```

- Typically, your search criteria will be a name, and you'll want to search your whole file system
- `$ find / -name [file_to_look_for]`
- For example, find the whoami program by typing:

```
$ find / -name whoami
```


The locate command is an efficient way to determine where files are located on the system. It consults a local database installed and updated by the system administrator for files that are frequently sought. It runs quickly and doesn't consume a lot of resources. However, it cannot locate items that are not loaded into its database.

To try locate, type:

```
$ locate whoami
```

If your system complains that there isn't a locate database or that it's out of date, you can manually update the database by typing the command:

```
# updatedb
```

To do a comprehensive search of the directory, you can use the find command. This command consumes a lot of resources. Several finds running simultaneously will slow a Linux system to a crawl. Still, find is the best way to find something if locate doesn't work.

Let's try to find a file on the file system. Type the following:

```
$ find / -name whoami
```

What do you see?

© 2016 Ed Skoudis, John Strand

Editing Files (gedit)

- There are several editors included in most Linux variants:
  - vi, gnu-emacs, pico, mcedit, gedit
- For new users, gedit is easy to learn
  - Although easy to use, it's powerful

```
$ gedit [filename]
```

... as in ...

```
$ gedit test_file
```


You may need to edit a file at some point. You can use any editor you are comfortable with. If you are new to Linux, you should consider using gedit, one of the easiest editing tools commonly installed in Linux. If you have a GUI, you can use gedit.

Let's create and edit a file:

```
$ cd ~                    (change to the home directory)
$ gedit test_file         (let's edit and create a file named "test_file")
```

Now, edit your file. Type in a bunch of junk. Use the function keys to save it.

I told you gedit was easy!

© 2016 Ed Skoudis, John Strand

Viewing File Contents (cat, head, and tail)

- The cat command shows the contents of a file

```
$ cat /etc/passwd
```

... or:

```
$ cat ~/test_file
```

- The head command shows the start of a file
  - 10 lines by default
  - Or specify -n [n] for seeing first n lines

```
$ head /etc/passwd
```

- The tail command shows the end of a file
  - Again, 10 lines default, or -n [n]

```
$ tail -n 2 /etc/passwd
```


So, you just edited a file. How can you see its contents? You can use the cat command:

```
$ cat ~/test_file
```

Also, you can look at other files:

```
$ cat /etc/passwd
```

This shows the contents of the password file! (Note that on most Linux installations, the passwords are stored in another file, /etc/shadow). Typically, in most modern UNIX installations, /etc/passwd just contains account information.

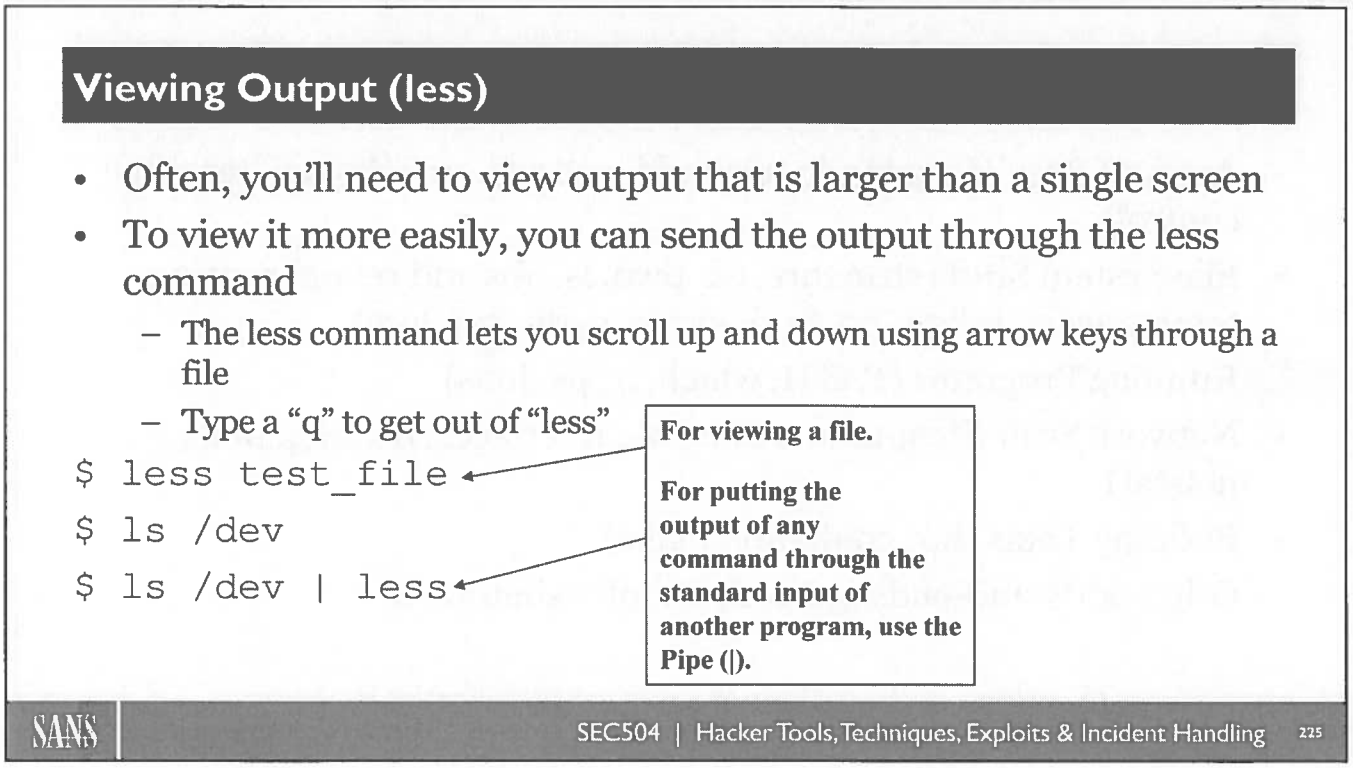
Alternatively, we can view portions of files using the head or tail commands. The head command shows the first 10 lines of a file by default. By specifying head -n [n] [filename], we can view just the first n lines. Similarly, the tail command shows the last 10 lines of a file by default, or we can use the -n [n] syntax to view a different number of trailing lines. Consider the following commands:

```
$ head /etc/passwd
```

```
$ head -n 1 /etc/passwd
```

```
$ tail -n 2 /etc/passwd
```


© 2016 Ed Skoudis, John Strand



In addition to cat, there are other commands you can use to look at files. The less command is one of the best to use. Try typing:

```
$ less test_file
```

You should see the contents of the file.

In addition to looking at files, the less command can also be used to help look at lengthy output from a command. Try typing:

```
$ ls /dev
```

This shows you all the devices (virtual and otherwise) on your system. It's a long, unwieldy list. The less tool lets you interact with this output in a better way.

Type:

```
$ ls /dev | less
```

By piping the output of ls through less, you can now use the cursor keys to scroll up and down through the output. The space key jumps forward one page. Use the "q" key to quit. The pipe takes the output of one program and feeds it into the standard input of another program.

© 2016 Ed Skoudis, John Strand

Intro to Linux Topics

- Account Stuff (Logging In, useradd, passwd, su, whoami, terminal control)
- File System Stuff (structure, cd, pwd, ls, abs and rel referencing, mount, eject, mkdir, cp, find, locate, gedit, cat, less)
- Running Programs (PATH, which, ./, ps, jobs)
- Network Stuff (ifcfg-eth0, restarting interfaces, ifconfig, ping, netstat)
- Building Tools (tar, configure, make)
- Other odds-and-ends (grep, man, info, shutdown)

You guessed it ... another outline slide. We will now discuss running programs. This section is important (not that the other ones aren't important ...). People frequently mess up on this stuff and get confused because Linux works differently from Windows in running programs.

© 2016 Ed Skoudis, John Strand

Running Programs (PATH and which)

- You can type a program's name at the command prompt to run the program
- When you type a program name on the command line, it looks for the program in your path
- The path is established using the environment variable $PATH
- View your path by typing:

```
$ echo $PATH
```

- You can see where your commands are run from by using the which command

```
$ which ls
```


When you type a command at the prompt, the system looks in your PATH to find the right program to run.

Look at your path by typing:

```
$ echo $PATH
```

The echo command means "type the following." The $ before path means, "What follows isn't a string of characters; it's a variable." The variable we want to type is our PATH.

The result is a list of directories where the system searches for programs based on what we type at the command line. These directories are separated by a ":". If you type a program name at a command prompt, and the program isn't in your PATH, the system will tell you that it cannot find the program. You have to either refer to it absolutely or relatively, or add its directory to your PATH.

If you want to see where in your PATH a command has been found, you can use the which command. Try typing:

```
$ which ls
```

That's where your "ls" program really is!

© 2016 Ed Skoudis, John Strand

Running Programs Not in Your PATH

- Note that the current directory "." is not in your path!
  - This is good because then you cannot be tricked into running a Trojan Horse
  - Think what would happen if I created a backdoor named ls
- So, how do you run a program if you are in the current directory of that program?
- Use relative referencing, from "."

```
$ ./[program_name]
```

- Or just use absolute referencing

This is an important point that confuses people, because UNIX functions differently from Windows on this issue.

For security reasons, your current working directory (the one shown by "pwd"), also referred to as ".", is not in your PATH. That's a good thing! If "." were in your path, an evil attacker could name an evil Trojan Horse program ls and put it in your home directory. When you ran ls to look at your home directory's contents, you'd run the evil trojan horse! For this reason, "." isn't in the path by default and shouldn't be put in your path.

This also means that if you change directories to a place in which a program file is located, you cannot just type the program's name to run it. Instead, to run the program, you have to type ./[program_name] to run it.

If the system ever complains that it cannot find a file but you can see the file in the current working directory using ls, you likely just need to start the program by typing:

```
$ ./[program_name]
```

On Windows machines, the current working directory is in your path. Therefore, if you change to a directory with an executable and type the executable's name on Windows, the program runs. Yes, it's convenient .... However, it's a security hole!

© 2016 Ed Skoudis, John Strand

Adding Directories to Your PATH

- To add directories to your path temporarily:
  - Temporarily means just for a given terminal session and processes started from it:

```
$ PATH=$PATH:[another_dir]
```

- To change your path permanently for this account, you must edit the .bash_profile file
  - I advise you to avoid doing this

Although we DO NOT recommend it, you could add a directory to your PATH temporarily. Type the following:

```
$ echo $PATH
```

Look at your path. To change your path temporarily, you could type (NOT RECOMMENDED):

```
$ PATH=$PATH:/[another_directory]
```

Now, type:

```
$ echo $PATH
```

Your path will now include the additional directory at its end.

This change applies only to this terminal and any processes started from this terminal. When you logout, this change goes away, and your path has its original settings.

To permanently change your path, you must edit the ~/.bash_profile file. I advise you to avoid editing this file if you are new to Linux. The default path setting is good for most purposes.

© 2016 Ed Skoudis, John Strand

Looking at Running Processes (ps)

- The ps command shows you processes running on the machine (sort of like the Windows Task Manager)

```
$ ps aux
```

- Or better yet:

```
$ ps aux | less
```

- Columns show process user, PID, CPU and Memory Utilization, Start Time, Time Running, and Command Line Invocation
- Or use the top command, which is even more like Task Manager (continuously updated):

```
# top
```


Sometimes, you need to see what processes are running. The ps command shows you a bunch of info about all running processes. Try typing:

```
$ ps aux
```

You get an exhaustive (and exhausting) list of all running processes.

Let's use our little "less" trick to make this output more readable:

```
$ ps aux | less
```

Now, you can scroll up or down and get a better feeling for what's running on your system.

© 2016 Ed Skoudis, John Strand

Job Control: CTRL-Z and bg

- At a single command prompt, you can run and control multiple programs simultaneously
- Execute a program, such as:
  `$ find / -name ls`
- Terminate the program by pressing CTRL-C:
- Now, run it again:
  `$ find / -name ls`
- Stop (Pause) the program by pressing CTRL-Z
- To start the program again in the background, type:
  `$ bg`
- So, you've just gotten your shell back while the program continues to run in the background!

You can temporarily pause programs with CTRL-Z and get your command prompt back. This is quite useful because you can get your command prompt back to run more programs if you want.

Also, you can restart the paused program running in the background with the bg command. The fg command starts it running in the foreground, as you might expect.

Let's try it. Type:

```
$ find / -name ls
```

Before it finishes running, press `CTRL-Z`.

Now, restart the program in the background by typing:

```
$ bg
```

© 2016 Ed Skoudis, John Strand

More Job Control: &, jobs, and fg

- Alternatively, you can run a program and send it to the background right away by using &
  ```
  $ find / -name ls &
  ```
- You can run a whole bunch of programs in the background this way
- To get a list of programs you have running in the background, use the jobs command
  ```
  $ jobs
  ```
- To bring one of the jobs into the foreground, type fg and the job number
  ```
  $ fg 1
  ```
- The default for fg is the most recent job sent to the background

The jobs command gives you a list of all programs you have kicked off that are running in the background. The fg command can also be used to restart a specific paused program in the foreground, but giving the job number after the fg command.

If the find command from the previous slide has finished, type the same command again, but this time run it in the background using the & after the command invocation.

```
$ find / -name ls &
```

As it runs in the background, type the jobs command:

```
$ jobs
```

Look at the job running. You can move it to the foreground by typing:

```
$ fg 1
```


© 2016 Ed Skoudis, John Strand

Intro to Linux Topics

- Account Stuff (Logging In, useradd, passwd, su, whoami, terminal control)
- File System Stuff (structure, cd, pwd, ls, abs and rel referencing, mount, eject, mkdir, cp, find, locate, gedit, cat, less)
- Running Programs (PATH, which, ./, ps, jobs)
- Network Stuff (ifcfg-eth0, restarting interfaces, ifconfig, ping, netstat)
- Building Tools (tar, configure, make)
- Other Odds-and-Ends (grep, man, info, shutdown)

Another outline slide. Gee, these outlines are fun.

Now, we will cover getting and staying networked in Linux.

© 2016 Ed Skoudis, John Strand

Setting Up Linux Networking

- Edit the network config using your favorite editor, such as gedit, mcedit, vi, or emacs

```
# gedit /etc/network/interfaces
```

- In this file, you can set interfaces to static or dhcp
- And, you can set specific IP addresses, netmasks, and more

To set your network interface options in Linux, you can edit the /etc/network/interfaces file. By putting in the appropriate information, you can configure your interface for static addresses or dhcp.

© 2016 Ed Skoudis, John Strand

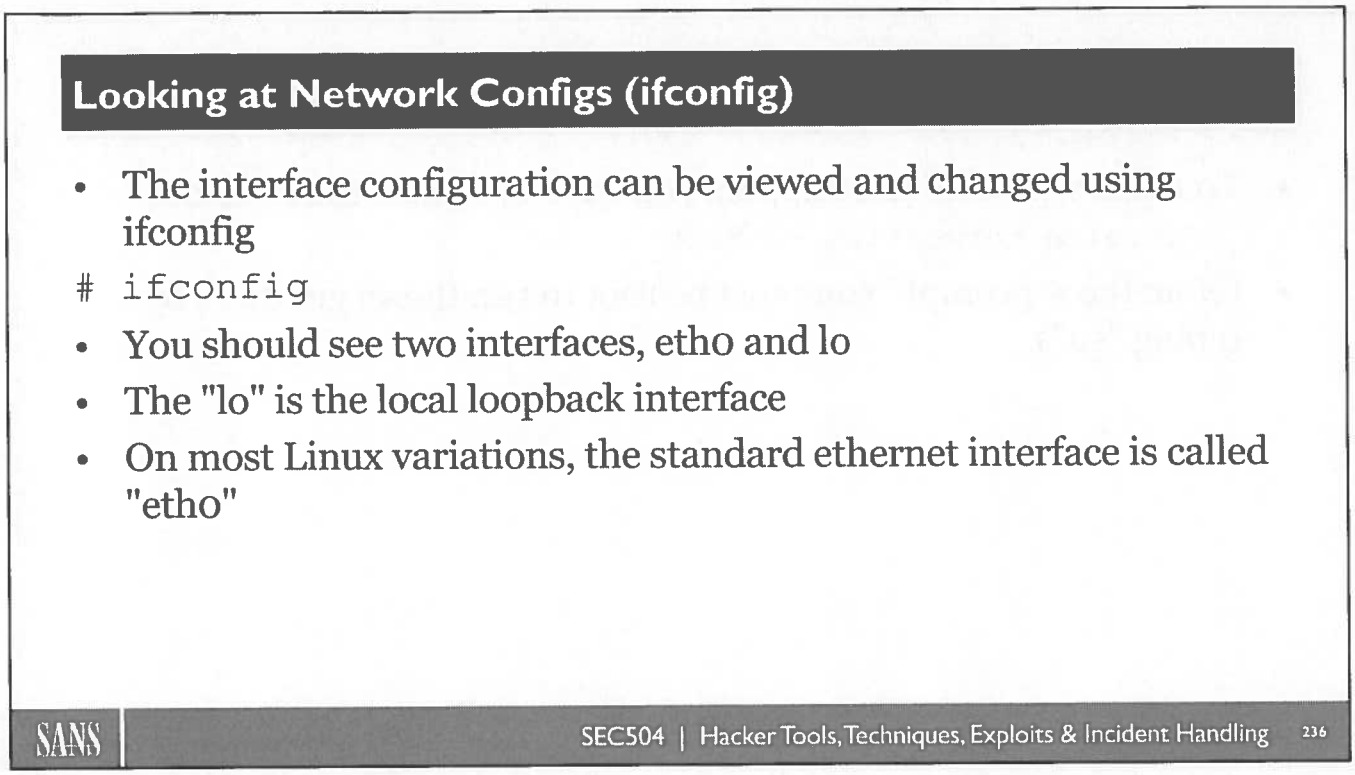
Applying Network Config Changes (Restarting Interfaces)

- To make your changes happen, you have to restart the interface
  `# service networking restart`
- (Note the # prompt! You must be root to run these; get there by typing "su")

If you change the interfaces file, your changes will not be applied to the interface immediately. Instead, you need to restart your interface. Type (as root):

```
# service networking restart
```

© 2016 Ed Skoudis, John Strand



Let's see if our interface changes were applied to the system. To look at your interface configuration, type:

```
# ifconfig
```

You see your IP address, netmask, MAC address, and various other nifty items. If you have one ethernet card, you see two interfaces" the local loopback interface with the address 127.0.0.1 and your ethernet interface, called eth0.

© 2016 Ed Skoudis, John Strand

Pinging (ping)

- Ping sends ICMP Echo Request messages to another host and prints out whether it gets a response
- You can use it to verify that you are properly networked

```
$ ping [IP_Address]
```

- Press CTRL+C to stop it

To verify that you are properly networked, you can ping another machine. The ping command is similar, but not identical, to the Windows ping program. One of the biggest differences is that a Linux ping keeps sending pings until you press <CTRL+C> to stop it. By default, the Windows ping sends out four ICMP Echo Request packets and then stops. Linux just keeps going until you stop it.

© 2016 Ed Skoudis, John Strand

Looking at Network Usage (Netstat)

- The Netstat command shows information about the system's network interfaces
- It can show routing tables, current connections, and listening ports
- We will use it to show listening ports:

```
$ netstat -nap
```

- Or better yet:

```
$ netstat -nap | less
```

- Look for "LISTENING" and "ESTABLISHED"
- Alternatively, you can use the lsof command:

```
$ lsof -i | less
```

SANS | SEC504 | Hacker Tools, Techniques, Exploits & Incident Handling 238

Now look at what's using your various TCP and UDP ports. Type:

```
$ netstat -nap
```

There's a lot of stuff there. It can be a bit difficult to read as it scrolls by, so try this:

```
$ netstat -nap | less
```

You can scroll up and down through the output. We'll discuss how to do better searches through this later.

Note that various TCP and UDP ports are shown as LISTENING. These are waiting for a connection. Others may indicate that they are ESTABLISHED. These have existing connections.

© 2016 Ed Skoudis, John Strand

Intro to Linux Topics

- Account Stuff (Logging In, useradd, passwd, su, whoami, terminal control)
- File System Stuff (structure, cd, pwd, ls, abs and rel referencing, mount, eject, mkdir, cp, find, locate, gedit, cat, less)
- Running Programs (PATH, which, ./, ps, jobs)
- Network Stuff (ifcfg-eth0, restarting interfaces, ifconfig, ping, netstat)
- Building Tools (tar, configure, make)
- Other Odds-and-Ends (grep, man, info, shutdown)

You know, to use Linux in the workshop, you have to run tools. To run them, in many cases you need to build and install them. As you see from the outline slide, we'll cover building and installing tools next.

Some programs are installed by using tar files. Others are rpms. Still others use "configure" and "make." How do you know which tools use which format? Most of the tools include a README file. Look at the README file (using cat, less, or gedit) for instructions on installing the tool. The course notes for the main class also include directions for compiling and installing.

By the way, we have you compile and install the tools so that you can get experience with doing these tasks. In the wild, you may need to compile and install new versions of these and other tools, so we want to get you ready.

© 2016 Ed Skoudis, John Strand

Untarring Files in Linux (tar)

- If the file format ends in .tar, it is a tape archive image
  - To untar it, type:

```
$ tar xvf [archive.tar]
```

- If the file format ends in .tar.gz or .tgz, it is a compressed tape archive image
  - To uncompress and untar it, type:

```
$ tar xvfz [archive.tar.gz or archive.tgz]
```


Some tools are stored at tape archives, abbreviated "tar." This doesn't mean they were on physical tapes; the lingo just lingers from the olden days. Although tapes may or may not be used, tar files are used all the time. Think of them as being like ZIP archives in Windows. You take a bunch of files and glom them together in a tar file.

To open a tar file, you use the xvf parameters. x means "Extract." v means "Be verbose; give me a lot of output to let me know what's going on." f means "Get this from a file."

If the tar file has been compressed using a tool called "gzip," its name will end with a suffix of .tar.gz or simply .tgz. To open these, you need to use the tar command with the xvf and z flags. The z flag means "Unzip this before you open the archive."

When the archive opens, all files and directories associated with it will be automatically created in the current working directory and below.

We won't demo this during the Intro to Linux mini-workshop. You'll get a chance to use tar files during the main class.

© 2016 Ed Skoudis, John Strand

Building Linux Tools: Configure and Make

- Some tools are not precompiled
- A script is included with some tools to properly configure your system
- The make program then compiles it
- The make install command then installs the components
- So, for these tools, you must do the following:

```
$ ./configure
$ make
$ make install
```


Although some programs ship as tar files and others as RPMs, many just ship with a script called "configure." You need to run this script first, which checks your environment and creates a set of options necessary to get the tool compiled on your device. After running configure, you run the make command, which compiles and builds the tool. Then, by typing **make install**, the program is loaded into the appropriate place.

We won't demo this during the Intro to Linux mini-workshop. You'll get a chance to use configure and make during the main class.

© 2016 Ed Skoudis, John Strand

Building Linux Tools: Make

- For some tools, there is no configure script
- You simply use the make program to compile it

```
$ make
```


Some tools don't have a "configure" script. For these, you just run the `make` command.

© 2016 Ed Skoudis, John Strand

Intro to Linux Topics

- Account Stuff (Logging In, useradd, passwd, su, whoami, terminal control)
- File System Stuff (structure, cd, pwd, ls, abs and rel referencing, mount, eject, mkdir, cp, find, locate, gedit, cat, less)
- Running Programs (PATH, which, ./, ps, jobs)
- Network Stuff (ifcfg-eth0, restarting interfaces, ifconfig, ping, netstat)
- Building Tools (tar, configure, make)
- Other Odds-and-Ends (grep, man, info, shutdown)

Here are some other odds and ends that can help us use Linux throughout this course.

© 2016 Ed Skoudis, John Strand

Sorting through a Bunch of Data (grep)

- The grep command finds items that match a given condition
- To find files in the current directory that contain the word root, type:

```
$ grep root *
```

- Read this as grep for the string root from star
- The * means all files in this directory

Grep is a powerful tool for finding data. It can look through files or the output from commands to identify particular strings. We will just scratch the surface of its use.

To look for a given string in a set of files in a directory, you can type:

```
$ grep root *
```

This prints all occurrences of the word "root" and the file in which it appears in the current working directory. Try the following:

```
$ cd /etc
```

```
$ grep root *
```

See the word "root" in any files here? Which ones?

© 2016 Ed Skoudis, John Strand

Using grep with netstat and ps

- To see if anything is listening on port 7777, you could type:

```
$ netstat -nap | grep 7777
```

- To see if you have any processes named bash running, you could type:

```
$ ps aux | grep bash
```


Grep can help isolate information about the usage of particular ports and processes.

At the command prompt, type:

```
$ netstat -nap | grep 7777
```

This says, "Run the netstat command to show me TCP and UDP port usage, send the output to grep, and have grep show me any lines with the string 7777 in it." The results indicate if anything is listening on or using port 7777.

Likewise, you can use grep to help you find particular programs. At a command prompt, type:

```
$ ps aux | grep bash
```

This shows you all processes running the bash program (the command shell you are running) that are currently executing on your system.

© 2016 Ed Skoudis, John Strand

To Learn More (man and info)

- The man and info commands show detailed usage information for other commands, for example:

```
$ man ls
$ info ls
$ man man
```


To learn more about Linux, you can use man or info.

Try the following:

```
$ man ls
```

Interesting ... and chilling. The `ls` command is complex!

Also, try:

```
$ info ls
```

And, check this out to learn more about man:

```
$ man man
```


© 2016 Ed Skoudis, John Strand

Getting Hints from whatis and apropos

- If you don't want to look through an entire man page, and just need a hint about what a program does ...

```
$ whatis [command]
```

- As in:

```
$ whatis ifconfig
```

- You can also use the apropos command to search for topics:

```
$ apropos network
```

  - This is the equivalent of man –k to look up something by keyword, as in:

```
$ man -k network
```


The whatis command is useful for getting hints from the system about what various commands do. It won't change your life, but it might just jog your memory about some esoteric command.

I usually just use the man page, but some people prefer whatis.

Try typing:

```
$ whatis ifconfig
```

You can also use the apropos command to search for topics and the commands related to those topics:

```
$ apropos network
```

This is the equivalent of man –k to look up something by keyword, as in:

```
$ man -k network
```

© 2016 Ed Skoudis, John Strand

Shutting Down (Shutdown and Reboot)

- You can do this via the GUI ...
- ...or at a command line
- To shut down and halt the system, type (as root):

```
# shutdown -h now
```

- To shut down and reboot the system, type (as root):
  - `# shutdown -r now`
  - Or just type:

```
# reboot
```


When you are done with Linux, you should shut it down gracefully.

You can do this from the GUI, but I usually just do it from the command prompt.

As root (you may need to su!), to gracefully shut down your system, type:

```
# shutdown -h now
```

The -h flag means "halt" the system. Of course, "now" means do it right away. You can actually schedule the system to shutdown at another time using this command, too.

You can also use the shutdown or reboot command to reboot the machine. To reboot, I usually just type:

```
# reboot
```


© 2016 Ed Skoudis, John Strand

Intro to Linux: Conclusions

- You have the building blocks you need to participate in the full class
- Linux is powerful but sometimes frustrating
- Refer to this section during the main class
- Ask for help from instructor/proctors/mentors if required

You are now ready for the full class Linux labs! Use your new-found Linux skills for good, not evil!

© 2016 Ed Skoudis, John Strand

Course Roadmap

- **Incident Handling**
- **Applied Incident Handling**
- **Attack Trends**
- **Step 1: Reconnaissance**
- **Step 2: Scanning**
- **Step 3: Exploitation**
  - Gaining Access
  - Web App Attacks
  - Denial of Service
- **Step 4: Keeping Access**
- **Step 5: Covering Tracks**
- **Conclusions**

APPLIED INCIDENT HANDLING

1. Espionage
2. Unauthorized Use
3. Insider Threats
4. Intellectual Property Attacks
5. Legal Issues and Cyber Crime Laws
   - Lab: Analyzing the Evil Insider
7. Appendix A: Intro to Linux Workshop
8. **Appendix B: Lab: Linux Cheat Sheets**

We now discuss the Linux cheat sheet, which contains a set of instructions that an administrator can run to help determine if a machine has been compromised.

© 2016 Ed Skoudis, John Strand

Linux Cheat Sheet – Unusual Processes

- Look for running processes
  `# ps aux`
- Get familiar with "normal" processes
- Look for unusual processes
  - Focus on processes running with root privileges
- Investigate unusual processes, getting more detail by using
  `# lsof -p [pid]`
  - Shows all files and ports used by running process

First, we need our sys admins to look for unusual processes running on a machine. These processes could represent a worm, a backdoor, a sniffer, a password cracker, or some other nefarious activity of the bad guy.

On Linux, there are many methods of looking at process activity. One of the most straightforward is simply running the ps command with the aux flags (note that the "-" in front of those flags is optional on most modern Linux systems). To spot an anomaly, however, our administrators need to know what kind of processes they should normally see on their systems.

We have them look for processes that they don't expect to be there, especially those running with root privileges.

When they find an unusual process, system administrators can use the "lsof –p [pid]" command to show all files and TCP/UDP ports used by the process. The lsof command is built into most modern Linux distributions and is immensely helpful in determining the status of the machine.

© 2016 Ed Skoudis, John Strand

Linux Cheat Sheet – Unusual Services

- On machines where it's installed, service is helpful
  - RedHat and Mandrake equivalent is chkconfig, available for others
- Can be used to look at and modify start-up configuration for services initiated through rc.d and xinetd
  - Shows on/off status for each runlevel
  - We don't have to edit the rc.d or xinetd.d directories any more
- **# service --status-all**

When looking for unusual services, the chkconfig command is helpful. It's built into many Linux distributions, especially those that look and smell like Red Hat (Red Hat itself, Mandrake, etc.).

Use the **status-all** switch to list all services that are registered with the OS and issues them a **status** command. You will then get one of the following displayed next to each service:

[+]: Services with this sign are currently running

[–]: Services with this sign are not currently running

[?]: Services that do not have a **status** switch

Chkconfig, when used with the "--list" flag, shows the on or off status of various services started through the rc.d files and xinetd. It'll show whether the service is activated at each particular run level. For many recent Linux distros (including Red Hat 8.0 and later), the default boot is to runlevel 5. You can check your current runlevel by typing the "runlevel" command. Then, run "chkconfig –list" to see what is on or off for various run levels.

Chkconfig can also be used to turn on or off various services, so you don't have to edit rc.d or xinetd.d directories or scripts any more.

There's even a version of chkconfig for Solaris, in case you are interested.

© 2016 Ed Skoudis, John Strand

Linux Cheat Sheet – Unusual Files

- Look for unusual SUID root files
  ```
  # find / -uid 0 -perm -4000 -print
  ```
  – Requires knowledge of normal SUID files
- Look for unusual large files (greater than 10 MB)
  ```
  # find / -size +10000k -print
  ```
  – Requires knowledge of normal large files
- Look for files named with dots and spaces ("...", ".. ", ". ", and " ")
  ```
  # find / -name " " -print
  ```


Next, we have the administrator look for unusual files. First, we have him search for SUID root files, which are scripts and binaries that run with the permissions of their owner, instead of the permissions of the person running the program. Attackers sometimes leave SUID root files on a system to act as a backdoor to root-level access. Of course, to spot unusual SUID files, the system administrator needs to know what files normally have SUID root permission on the machine.

Additionally, to look for attackers that store large archives of stolen software, password lists, or pornography, we have the administrator run the "find" command to look for files larger than 10 MB. Again, the administrator should have a feel for which files should normally be that large.

Attackers often create files with a name that has a space (" "), three-dots ("..."), a dot-space (". "), or any other of a myriad of possibilities. This technique is common enough that diligent system administrators should periodically search for such files using the commands on this slide.

© 2016 Ed Skoudis, John Strand

More Unusual Files

- Sometimes, an attacker runs a backdoor or stashes some data on a machine...
- ... and unlinks it so that it doesn't appear in the normal directory structure
- We can find such files using lsof

```
# lsof +L1
```

- Shows files with a link count less than 1 (that's 0 to you and me)

Sometimes, an attacker runs a backdoor program and then unlinks the executable file associated with the backdoor. The backdoor continues to run, but its binary cannot be easily spotted with the "ls" command. This is a fairly common trick among certain bad guys.

The "lsof" command comes to the rescue here. We can run "lsof +L1" to show all processes that have open files with a link count less than 1. A link count of less than 1 is, of course, 0... an unlinked, orphaned file floating through the file system. It's a good idea to periodically look for such items on the machine.

We do a lab later that tests this action.

© 2016 Ed Skoudis, John Strand

Even More Unusual Files

- On a Debian Linux machine, run the debsums tool to verify packages
  - **# debsums**
  - Checks size, MD5 sum, permissions, type, owner, and group of each file with information from package database
  - Pay special attention to changes associated with items in /sbin, /bin, /usr/sbin, and /usr/bin

```
sec504@slingshot:~$ debsums
/lib/systemd/system/accounts-daemon.service                    OK
/usr/lib/accountsservice/accounts-daemon                       OK
/usr/share/dbus-1/interfaces/org.freedesktop.Accounts.User.xml OK
/usr/share/dbus-1/interfaces/org.freedesktop.Accounts.xml      OK
/usr/share/dbus-1/system-services/org.freedesktop.Accounts.service OK
/usr/share/doc/accountsservice/README                          OK
```


Finally, on a Debian Linux system equipped with the *debsums* tool, which searches the package database and verifies the MD5 checksum. This simple command can spot some of the most common changes made by attackers.

We request that administrators focus on changes in the bin directories, such as /sbin, /bin, /usr/sbin, and /usr/bin.

© 2016 Ed Skoudis, John Strand

Unusual Network Usage

Linux Cheat Sheet

- Look for unusual port listeners
  ```
  # lsof -i
  # netstat -nap
  ```
- Need to know which TCP and UDP ports are normally listening on your system and look for deviations from the norm
- Look for unusual ARP entries, mapping IP address to MAC addresses that aren't correct for the LAN
  ```
  # arp -a
  ```
  – Requires detailed knowledge of what is supposed to be on the LAN

The Linux cheat sheet includes instructions for looking for listening TCP and UDP ports, using the "lsof –i" and "netstat –nap" commands. Both of these commands show the process name listening on the port, which indicates to the sys admin what is going on.

Finally, the sys admin can check the Address Resolution Protocol (ARP) entries of the machine to look for unusual mappings of IP address to MAC address. Unfortunately, interpreting these results and looking for anomalies requires the administrator to have detailed knowledge of the IP and MAC addresses that are supposed to be on the LAN.

© 2016 Ed Skoudis, John Strand

Unusual Scheduled Tasks | Linux Cheat Sheet

- Look for cron jobs scheduled by root and any other UID 0 accounts

```
# crontab -l -u root
```

- Look for unusual system-wide cron jobs

```
# cat /etc/crontab
# ls /etc/cron.*
```

This is a dash-lower-case-L, not a dash-one.

Beyond unusual files, we need administrators to look for cron jobs scheduled to run on the box. Attackers often schedule jobs that include backdoors to run on the machine, guaranteeing the attacker return access to the system.

First, we have the administrator look at root's crontab file by running "crontab –l –u root." Note that this command contains a dash-lowercase-L, not a dash-one.

The admin should also look for crontabs assigned to other UID 0 accounts.

Additionally, the Linux sheet describes how to analyze system-wide cron jobs by looking at the /etc/crontab file and then the individual files, /etc/cron.daily, /etc/cron.weekly, and so on.

© 2016 Ed Skoudis, John Strand

Unusual Accounts

Linux Cheat Sheet

- Look in /etc/passwd for new accounts, sorted from lowest to highest UID

```
# sort /etc/passwd -nk3 -t: | less
```

- Sorts by number, keys the sort on third column (UID), and uses a delimiter of ":"
- Also, look for UID and GID 0 accounts

```
# grep :0: /etc/passwd
```

- Normal accounts will be there, but look for new, unexpected accounts

We also need to be diligent about accounts on the machine, especially those with rootly privileges (UID 0). First, we have system administrators run the /etc/passwd file through the sort command with the syntax: "sort /etc/passwd –nk3 –t: | less." This command sorts the password file numerically (n), sorting with a key of the third entry (k3) in a colon-delimited file (-t:). That way, we see the user accounts sorted by UID, with the most important ones at the top of the list.

Furthermore, by running "grep :0: /etc/passwd," admins find accounts with the string ":0:" to display UID 0 and GID 0 accounts. Of course, some normal accounts will have a UID or GID of 0. We need system administrators to look for new, unexpected accounts with these privileges.

© 2016 Ed Skoudis, John Strand

More Unusual Accounts

Linux Cheat Sheet

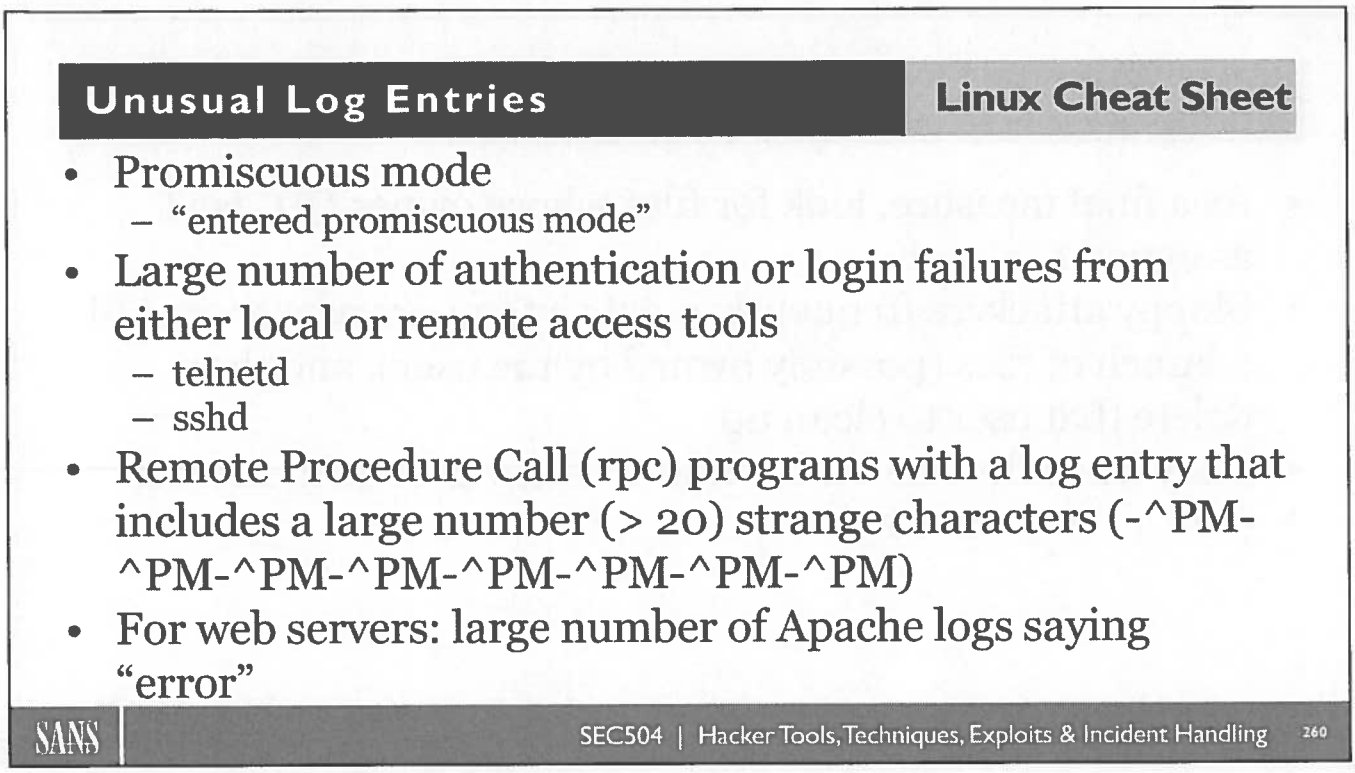
- As a final measure, look for files whose owner UID isn't assigned to a user
- Sloppy attackers frequently create a temporary user, install a bunch of files (possibly owned by the user), and then delete that user to clean up
- They leave the files with a non-existent user as the owner

```
# find / -nouser -print
```


As a final measure for unusual accounts and files, we search for files whose owner is a UID that's not currently defined on the machine. Sloppy attackers frequently create a temporary user, install a bunch of files (possibly owned by the user), and then delete that user to clean up. But, they leave the files with a non-existent user as the owner, giving us a chance to spot their nefarious deeds.

The find command, with the –nouser flag, helps us discover such treachery (as well as orphaned files, which is still nice to know about in keeping a system clean and secure!)

© 2016 Ed Skoudis, John Strand



Also, system administrators should periodically review their system logs, looking for signs of unusual behavior. Although thousands of different log entries could indicate an attack, some of the most telling issues include logs that indicate

- That an interface has entered promiscuous mode
- That someone has repeatedly tried unsuccessfully to log into the machine via telnet or sshd
- Remote Procedure Calls (rpc programs) being accessed with strange sequences of characters, which are possibly buffer overflows or format string attacks
- Large number of Apache error messages

© 2016 Ed Skoudis, John Strand

Other Unusual Items — Linux Cheat Sheet

- There are a few other areas to check for sudden changes in the resource utilization of the system
- First, the system load (CPU particularly)

```
$ uptime
```

- Next, the utilization of memory

```
$ free
```

- Finally, it's useful to check available hard-drive space

```
$ df
```


Because attackers often consume the resources of the target machine with their backdoors and such, we should check various aspects of the system (CPU, memory, and disk space) to look for sudden changes.

The administrator can check the CPU load by running the uptime command. The free command shows memory utilization. And, the df command shows the usage of disk space.

© 2016 Ed Skoudis, John Strand

Additional Supporting Tools

Linux Cheat Sheet

- Chkrootkit looks for anomalies on systems introduced by user-mode and kernel-mode RootKits
  - www.chkrootkit.org - free
- Tripwire looks for changes to critical system files
  - http://sourceforge.net/projects/tripwire/ - free for Linux for non-commercial use
- AIDE looks for changes to critical system files
  - http://sourceforge.net/projects/aide
- CIS Hardening Guidelines provide a starting point for secure system configuration
  - www.cisecurity.org - free
- Bastille hardening script to tighten security settings
  - http://bastille-linux.sourceforge.net - free

Finally, we finish the Linux cheat sheet with some additional tools that can be installed on a machine to spot evidence of attack. One of my absolute favorites is the free Chkrootkit tool, which looks for anomalies introduced by over 50 different user-mode and kernel-mode RootKits. It works like a champ on Linux (and several UNIX variants.)

Also, we mention Tripwire and AIDE, two file-integrity checking tools that can be used to fingerprint files and spot changes. Tripwire is free for Linux for non-commercial use at www.tripwire.org. The commercial version is available at www.tripwire.com. AIDE is free.

The CIS Hardening Guidelines provide a great starting point for secure system configuration. You can get them for free at www.cisecurity.org. Don't overlook the Bastille hardening script for Linux (and HP-UX and MacOS X). This tool tweaks the configuration of a system to shut off unneeded services and boost overall system security. It's free at www.bastille-linux.org.

Remember, if you don't want your admins to install or run these programs on your servers, remove this one item from the cheat sheet. When you photocopy the cheat sheet, just leave off the back panel, and you'll be in fine shape.

Of course, I really prefer to have Chkrootkit and Tripwire/AIDE on my box. It's awful hard to keep a system secure without the type of functionality they provide.

© 2016 Ed Skoudis, John Strand

Linux Lab

- Now, spend about 10 minutes running through some Linux lab steps
- Using Linux, we explore how the commands in these cheat sheets work
- For some elements, we just run the command
- For others, we create the unusual condition, and then show how the command detects it

Now, let's check out the cheat-sheet tips in action. You run several lab steps to look at the results your sys admins can expect to see. Boot up your Linux box.

For some of the cheat-sheet tips, we just run the command and look at our system's status.

For others, we actually create the condition the cheat-sheet tip is designed to detect. Then, we detect it using the tip. Then, we restore things, so that our box isn't left in a half-hacked state.

© 2016 Ed Skoudis, John Strand

Unusual Processes

- Run a Netcat listener
- Run ps to see all of the running processes
- Note the pid of the Netcat process and its command-line invocation
- Run "lsof –p [pid]" to zoom in on what Netcat is doing

```
# nc -l -p 2222 &
[1] 2456
# ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.5  22932  3052 ?        Ss   04:09   0:01 /sbin/init
root         2  0.0  0.0      0     0 ?        S    04:09   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        S    04:09   0:00 [ksoftirqd/0]
root         5  0.0  0.0      0     0 ?        S<   04:09   0:00 [kworker/0:0H]
root         7  0.0  0.0      0     0 ?        S    04:09   0:00 [rcu_sched]
root         8  0.0  0.0      0     0 ?        S    04:09   0:00 [rcu_bh]
root         9  0.0  0.0      0     0 ?        S    04:09   0:00 [migration/0]
root        10  0.0  0.0      0     0 ?        S    04:09   0:00 [watchdog/0]
root        11  0.0  0.0      0     0 ?        S<   04:09   0:00 [khelper]
root        12  0.0  0.0      0     0 ?        S    04:09   0:00 [kdevtmpfs]
root        13  0.0  0.0      0     0 ?        S<   04:09   0:00 [netns]
root        14  0.0  0.0      0     0 ?        S    04:09   0:00 [khungtaskd]
root        15  0.0  0.0      0     0 ?        S<   04:09   0:00 [writeback]
root        16  0.0  0.0      0     0 ?        SN   04:09   0:00 [ksmd]
root        17  0.0  0.0      0     0 ?        S<   04:09   0:00 [crypto]
root        18  0.0  0.0      0     0 ?        S<   04:09   0:00 [kintegrityd]
root        19  0.0  0.0      0     0 ?        S<   04:09   0:00 [bioset]
root        20  0.0  0.0      0     0 ?        S<   04:09   0:00 [kblockd]
root        22  0.0  0.0      0     0 ?        S    04:09   0:00 [kswapd0]
root        23  0.0  0.0      0     0 ?        S    04:09   0:00 [fsnotify_mark]
root        29  0.0  0.0      0     0 ?        S<   04:09   0:00 [kthrotld]
root        30  0.0  0.0      0     0 ?        S<   04:09   0:00 [ipv6_addrconf]
root        31  0.0  0.0      0     0 ?        S<   04:09   0:00 [deferwq]
root        63  0.0  0.0      0     0 ?        S<   04:09   0:00 [ata_sff]
root        65  0.0  0.0      0     0 ?        S    04:09   0:00 [scsi_eh_0]
root        68  0.0  0.0      0     0 ?        S<   04:09   0:00 [scsi_tmf_0]
```

This is a dash-lowercase-L, not a dash-one

First, let's look for unusual processes.

We'll start this simply by creating a copy of the Netcat program that can be used as a backdoor.

Run a Netcat listener on TCP port 2222 in the background:

```
# nc -l -p 2222 &
```
←*This is a dash-lowercase L, not a dash-one.*

Now, run the ps command to look at all of your processes, including Netcat. Check out the Netcat command line for invocation. Also, make a note of Netcat's pid number in the output of ps.

```
# ps aux
```

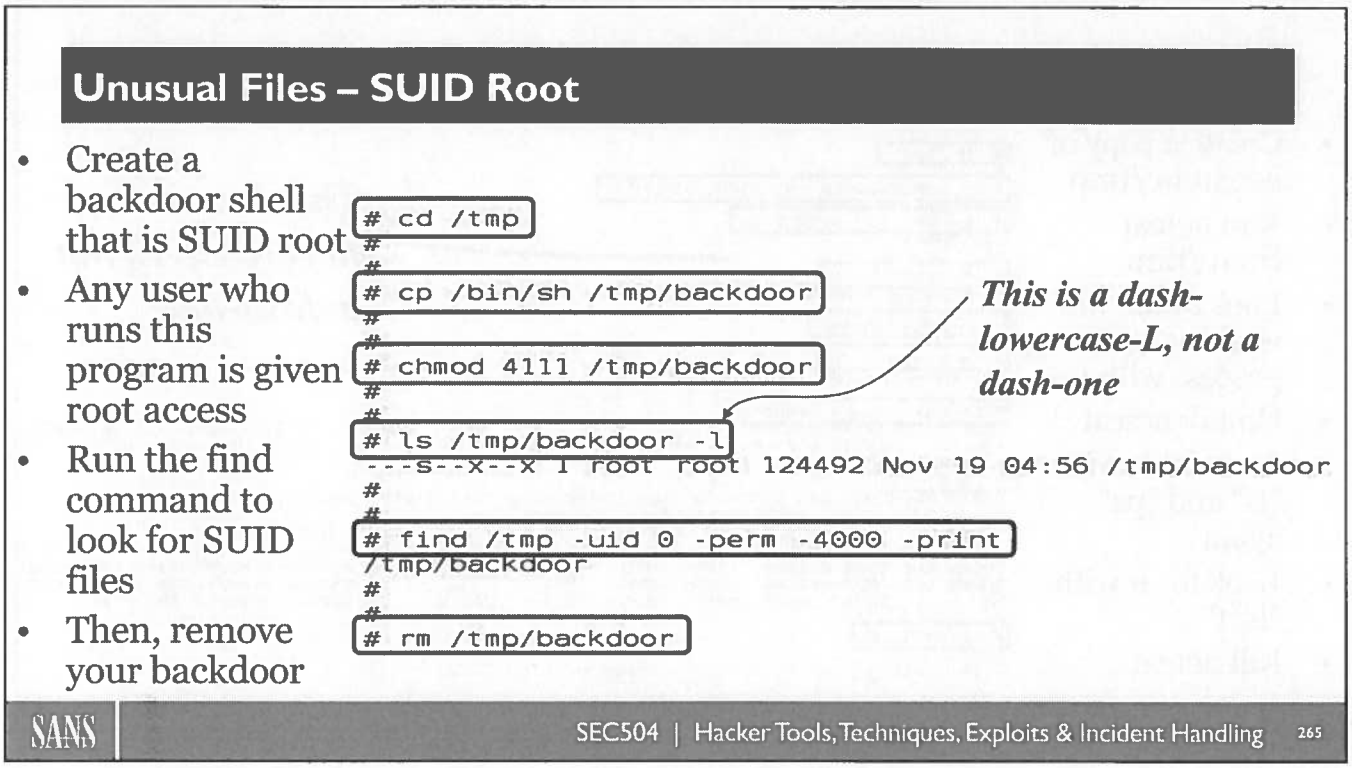
Then, run the lsof command to zoom into what files and ports the "nc" process has open.

```
# lsof -p [pid]
```

Remember, pid is the item you got from the ps command's output. Kill this little listener by running

```
# killall nc
```


© 2016 Ed Skoudis, John Strand



Now, let's turn our attention to unusual files. We'll start by creating and looking for SUID root files. These are the files that will always run with root privileges, regardless of who invokes them. Change into your /tmp directory:

```
# cd /tmp
```

Create a copy of the sh shell there, called backdoor:

```
# cp /bin/sh /tmp/backdoor
```

Next, let's set that backdoor so that it will execute, and will execute as its owner (SUID root):

```
# chmod 4111 /tmp/backdoor
```

Look at the permissions of this file by running

```
# ls /tmp/backdoor -l
```
←This is a dash-lowercase L, not a dash-one.

See the "s"? That means the program has the SUID bit set. Look at its owner: root. Now, if any user on the box with any permissions runs that backdoor file, that user is instantly given a root command prompt! That's a scary backdoor, giving all users on the box root if they find it. We can find it by running (from the cheat sheet):

```
# find /tmp -uid 0 -perm -4000 -print
```

Spot it? Bingo! Now, let's delete that dastardly backdoor:

```
# rm /tmp/backdoor
```

Then, type "y" to make it go away!

Unusual Files – Unlinked

- Create a copy of netcat in /tmp
- Run netcat from /tmp
- Look at the file with ls and the process with ps
- Unlink netcat
- Look for it with "ls" and "ps" again
- Look for it with "lsof"
- Kill netcat

```
# cd /tmp
#
# cp /home/tools/netcat/nc /tmp/nc
#
# /tmp/nc -l -p 2222 &
[1] 2364
#
# ls /tmp/nc -l
-rwxr-xr-x 1 root root 26216 Nov 19 05:10 /tmp/nc
#
# unlink /tmp/nc
#
# ls /tmp/nc -l
ls: cannot access /tmp/nc: No such file or directory
#
# ps aux | grep /tmp/nc
root      2364  0.0  0.2   2212  1472 pts/0    S    05:11   0:00 /tmp/nc -l -p 2222
root      2394  0.0  0.4   4536  2264 pts/0    S+   05:14   0:00 grep /tmp/nc
#
# lsof +L1
lsof: WARNING: can't stat() fuse.gvfsd-fuse file system /run/user/1002/gvfs
      Output information may be incomplete.
COMMAND    PID   USER   FD   TYPE DEVICE SIZE/OFF NLINK    NODE NAME
gnome-ter 1023 sec504   14u   REG    8,1    28672     0 402120 /tmp/#402120 (deleted)
gnome-ter 1023 sec504   16u   REG    8,1     4096     0 402143 /tmp/#402143 (deleted)
gnome-ter 1023 sec504   17u   REG    8,1     8192     0 402149 /tmp/#402149 (deleted)
nc        2364   root   txt   REG    8,1    26216     0 402121 /tmp/nc (deleted)
#
# killall nc
[1]+  Exit 1                  /tmp/nc -l -p 2222
```

This is a dash-lowercase-L, not a dash-one

Now, let's create a network-listening process and then unlink the executable associated with it. Bad guys do this sometimes to disguise their activities by making their backdoors harder to find. First, create a copy of netcat in the /tmp directory and run it:

```
# cd /tmp
# cp /home/tools/netcat/nc /tmp/nc
# /tmp/nc -l -p 2222 &
```
←This is a dash-lower-case L, not a dash-one.

Make sure you run netcat out of /tmp. If you just typed "nc" instead of "/tmp/nc," this part of the lab won't work! Now, look at the executable nc file:

```
# ls /tmp/nc -l
```

Then, unlink this file so that it won't show up with an ls:

```
# unlink /tmp/nc
```

Run ls again … nc is gone!

```
# ls /tmp/nc -l
```

Yet, netcat continues to run (as shown by ps), and even "ls –a" cannot see it.

```
# ps aux | grep /tmp/nc
# ls -a /tmp
```

But, our handy-dandy cheat sheet says to run this to see it:

```
# lsof +L1
```
(This is a plus-cap-L followed by a one!)

There it is! Kill it by running

```
# killall nc
```

Don't worry about that nc file that is unlinked. The system's garbage collection for the file system removes it.

© 2016 Ed Skoudis, John Strand

Network Usage – "lsof –i"

- Create a Netcat listener
- Run netcat in the back-ground
- Run "lsof –i" and look for your listener

This is a dash-lowercase-L, not a dash-one

```
# nc -l -p 2222 &
[1] 2442
#
# lsof -Pi
COMMAND     PID     USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
dhclient    391     root    6u  IPv4  10344      0t0  UDP *:68
dhclient    391     root   20u  IPv4  10315      0t0  UDP *:63925
dhclient    391     root   21u  IPv6  10316      0t0  UDP *:51859
sshd        468     root    3u  IPv4  11444      0t0  TCP localhost:22 (LISTEN)
avahi-dae   487    avahi   12u  IPv4  11462      0t0  UDP *:5353
avahi-dae   487    avahi   13u  IPv6  11463      0t0  UDP *:5353
avahi-dae   487    avahi   14u  IPv4  11464      0t0  UDP *:36455
avahi-dae   487    avahi   15u  IPv6  11465      0t0  UDP *:43225
nginx       521     root    6u  IPv4  11786      0t0  TCP *:80 (LISTEN)
nginx       521     root    7u  IPv6  11787      0t0  TCP *:80 (LISTEN)
nginx       522 www-data    6u  IPv4  11786      0t0  TCP *:80 (LISTEN)
nginx       522 www-data    7u  IPv6  11787      0t0  TCP *:80 (LISTEN)
nginx       523 www-data    6u  IPv4  11786      0t0  TCP *:80 (LISTEN)
nginx       523 www-data    7u  IPv6  11787      0t0  TCP *:80 (LISTEN)
nginx       524 www-data    6u  IPv4  11786      0t0  TCP *:80 (LISTEN)
nginx       524 www-data    7u  IPv6  11787      0t0  TCP *:80 (LISTEN)
nginx       525 www-data    6u  IPv4  11786      0t0  TCP *:80 (LISTEN)
nginx       525 www-data    7u  IPv6  11787      0t0  TCP *:80 (LISTEN)
```


Next, create a Netcat listener on TCP port 2222 again in the background:

```
# nc -l -p 2222 &
```
←*This is a dash-lower-case L, not a dash-one.*

Now, look for that little dude by running

```
# lsof -Pi
```

See it? What's its PID? How can you get more info about that process? (Hint: lsof is your friend.)

© 2016 Ed Skoudis, John Strand

Unusual UID 0 Accounts

- Look for UID 0 accounts with grep
- Create a new UID 0 account called test with useradd
- Look for UID 0 accounts again with grep

```
# grep :0: /etc/passwd
root:x:0:0:root:/root:/bin/bash
#
#
# useradd -o -u 0 -s /sbin/nologin test
#
#
# grep :0: /etc/passwd
root:x:0:0:root:/root:/bin/bash
test:x:0:1003::/home/test:/sbin/nologin
```


Now, we create and look for unusual accounts.

First, look for UID and GID 0 accounts on your box. There should be a few:

```
# grep :0: /etc/passwd
```

Now, let's add a new UID 0 (root-level!) account on the box. We'll run the useradd command, with the –o flag (to override the requirement that each account have a unique UID number), the –u flag (for a UID of 0), and the –s flag (to set the account's shell to /sbin/nologin, so no one can login with it). Name the account "test":

```
# useradd -o -u 0 -s /sbin/nologin test
```

Now, look for this account by running that grep against /etc/passwd again:

```
# grep :0: /etc/passwd
```

See it? I sure hope your sys admins would.

© 2016 Ed Skoudis, John Strand

Sorting Accounts

- Run the sort command and look for your test account
- Delete your test account when finished

```
# sort /etc/passwd -nk3 -t: | less

root:x:0:0:root:/root:/bin/bash
test:x:0:1003::/home/test:/sbin/nologin
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin

# userdel -r test
```


Now, let's run that "sort" command for /etc/passwd, so we can review all accounts, sorted by UID. That way, the UID 0 stuff appears at the top. Pay special attention to accounts with UIDs less than 500, because they are often associated with more sensitive information.

The syntax of our sort command includes "–n" (sort numerically), "k3" (key on the third entry of each line), and "-t:" (use a colon as the delimiter of the file).

```
# sort /etc/passwd -nk3 -t:  | less
```

Do you see your evil test account now?

To exit less, just press the Q key.

When you are finished with this page of the lab, delete the test account using

```
# userdel -r test
```

© 2016 Ed Skoudis, John Strand

Unusual Log Entries

- Run tcpdump to force the interface into promiscuous mode
- Look in /var/log/messages with grep to see promisc log entry

```
# tcpdump host 10.10.75.1 &
[1] 2183
# tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes

#
# grep promisc /var/log/messages
Nov 19 08:23:29 slingshot kernel: [ 1850.115843] device eth0 entered promiscuous
#
#
# killall tcpdump

0 packets captured
0 packets received by filter
0 packets dropped by kernel
[1]+  Done                    tcpdump host 10.10.75.1
```


As a final lab step, we look through our log files for an unusual entry associated with going into promiscuous mode.

First, run tcpdump in the background, and make it gather packets going to and from a host with IP address 10.10.75.1:

```
# tcpdump host 10.10.75.1 &
```

Next, look for a log entry that indicates that the interface went into promiscuous mode:

```
# grep promisc /var/log/messages
```

See it?

Good. Now, clean up after yourself by killing that sniffer:

```
# killall tcpdump
```

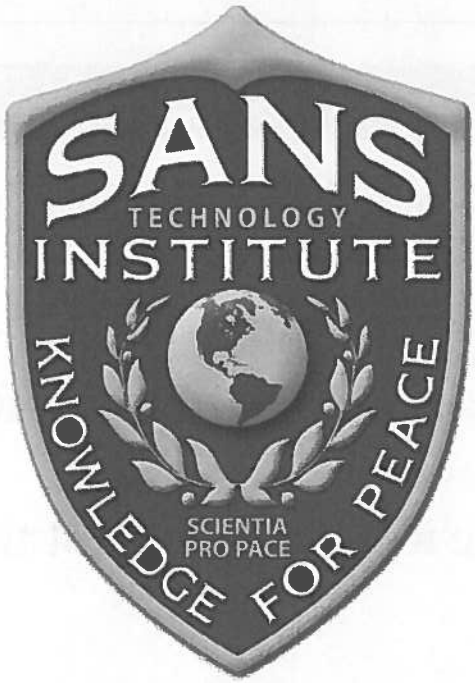

© 2016 Ed Skoudis, John Strand

Linux Cheat Sheets – The End

- That lab gave you a feel for using the Linux cheat sheet
- They have been an application of the various Linux tools we discussed during the Intro to Linux appendix
- We apply those ideas throughout the rest of this course

There you have it: The Linux cheat sheets are a direct application of what we covered in the Intro to Linux mini-workshop. We apply the concepts in the workshop and the cheat sheets throughout the rest of this course.

© 2016 Ed Skoudis, John Strand



This Course is Part of the SANS Technology Institute (STI) Master's Degree Curriculum.

If your brain is hurting from all you learned in this class, but you still want more, consider applying for a Master's Degree from STI. We offer two hands-on, intensive Master's Degree programs:

- Master of Science in Information Security Engineering
- Master of Science in Information Security Management

If you have a bachelor's degree and are ready to pursue a graduate degree in information security, visit www.sans.edu for more information.

www.sans.edu **855-672-6733** **info@sans.edu**

© 2016 Ed Skoudis, John Strand