

SANS

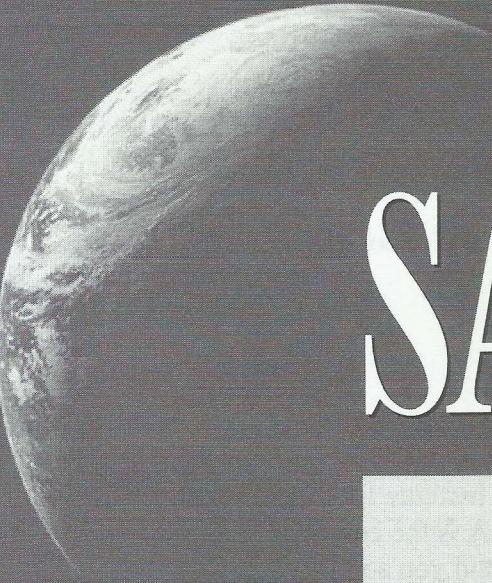
www.sans.org

SECURITY 503
INTRUSION DETECTION
IN-DEPTH

503.6

IDS Challenge

The right security training for your staff, at the right time, in the right location.



SANS

www.sans.org

SECURITY 503 INTRUSION DETECTION IN-DEPTH

503.6

IDS Challenge

The right security training for your staff, at the right time, in the right location.

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

IMPORTANT-READ CAREFULLY:

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

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

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

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

Sec503_6_A03

Intrusion Detection In-depth Roadmap

- 503.1: TCP/IP Refresher and Beyond
- 503.2: Introduction to Network Traffic Analysis
- 503.3: Advanced Network Traffic Analysis
- 503.4: Open Source IDS: Snort and Bro
- 503.5: Intrusion Analysis
- 503.6: IDS Challenge 

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

This is the day to apply all of the knowledge we stuffed in your brain for the week!

IDS Challenge

Analyze data from a
honeypot that gets
compromised multiple times

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

This page intentionally left blank.

Day 6 Roadmap

- Methodology Overview
- Unix Power Tools for Power Analysis
- Challenge Introduction
- Challenge Exercise
- Final Thoughts

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

Here is what your final day will entail. First, we'll discuss a general methodology to use to approach the Challenge. This is only a suggested method and if you find you would like to approach it differently, then go for it. While some of the Unix tools that may be handy for this Challenge were briefly discussed when we covered the log parsing exercise, we'll take some time to examine just what the commands are and how they can be used.

Then, we'll introduce and define the Challenge in more detail and allow you to do the exercise with peers, if you are in class, or alone if you are at home. We'll wrap up the day with some final thoughts about this week.

Methodology Overview

- Methodology Overview
- Unix Power Tools for Power Analysis
- Challenge Introduction
- Challenge Exercise
- Final Thoughts

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

This page intentionally left blank.

Methodology Overview

- Indicator or alert
- Drill down
- Pull reporting
 - Statistical
 - Queries
- Analysis write up

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

The basic flow of intrusion detection methodology will vary from organization to organization and from analyst to analyst. The basic components of that methodology should not change, however. Normally we get alerted of a suspicious event in the alerting/identification phase. This alert can come from a variety of sources we will discuss on the next slide. Following the alerting, the analyst will drill down using packet and log analyzers. Visualization can help get the big picture of the incident. Pull reporting gathers the disparate sources, ultimately using correlation of information, preparing the analyst for the final write up of the incident.

Indicator or Alert

- This is the first step in an incident
 - Snort alert
 - Log analysis finding
 - OSSEC, Splunk, SIM
 - Anti-virus or anti-malware alert
 - Help desk referral
 - Security reporting scripts
 - e.g. Looking for new admin accounts

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

The indicator or alert is the first indication that something is awry. This indicator or alert can come from an IDS, a log analysis system, anti-virus or anti-malware, reporting scripts, and even from help desk.

Snort Alert

- Which alerts should we investigate
- Start with highest priority, most likely to succeed against most critical target
- Running Snort against challenge.pcap

In our particular analysis, our first indicators for analysis will come from Snort.

We can run Snort against the pcap file for our Day 6 challenge (/home/sans/Exercises/Day6/challenge.pcap) with the command:

```
cd /home/sans/Exercises/Day6  
snort -c etc/snort.conf -K ascii -l log -r challenge.pcap
```

Just to give you an idea of the magnitude of what you are dealing with and the need to eliminate all but the most critical, find the total number of alerts:

From the "log" directory, run the command:

```
grep '\*\*\*' alert | wc -l
```

Sorting Alerts

```
212 [**] [1:402:8] ICMP Destination Unreachable Port Unreachable [**]
157 [**] [1:2923:9] NETBIOS SMB repeated logon failure [**]
126 [**] [1:399:6] ICMP Destination Unreachable Host Unreachable [**]
46 [**] [1:2050:15] SQL version overflow attempt [**]
46 [**] [1:2003:14] SQL Worm propagation attempt [**]
24 [**] [1:1394:12] SHELLCODE x86 inc ecx NOOP [**]
8 [**] [1:1882:14] ATTACK-RESPONSES id check returned userid [**]
6 [**] [1:408:5] ICMP Echo Reply [**]
5 [**] [1:2129:19] WEB-IIS nsiislog.dll access [**]
4 [**] [1:1243:20] WEB-IIS ISAPI .ida attempt [**]
1 [**] [1:542:14] CHAT IRC nick change [**]
1 [**] [1:498:7] ATTACK-RESPONSES id check returned root [**]
1 [**] [1:1887:5] MISC OpenSSL Worm traffic [**]
```

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

It is helpful in general, and more specifically, in this Challenge to sort the alerts you received. This is especially true when you have a large amount of data, as we do. This helps you summarize the issues that you need to analyze.

The alerts are sorted using some Unix text processing utilities, namely grep, sort, and uniq. The alerts will be sorted, unique ones only kept along with a count of all that specific type of alert, and sorted again in reverse numerical order to list the alerts that triggered most at the top of the listing and those that fired the least at the bottom.

From the "log" directory, run the command:

```
grep '\/*\*' alert | sort | uniq -c | sort -rn > sorted_alerts
```

A high number associated with the times the alert triggered doesn't necessarily mean that those are the ones of most interest. As you can see there are quite a few ICMP unreachable messages that truly don't interest us. In fact, these are so meaningless, that the Snort administrator ought to delete them from the rules directory/files.

Eliminating Alerts

```
212 [**] [1:402:8] ICMP Destination Unreachable Port Unreachable [**]
157 [**] [1:2923:9] NETBIOS-SMB repeated logon failure [**]
126 [**] [1:399:6] ICMP Destination Unreachable Host Unreachable [**]
46 [**] [1:2050:15] SQL version overflow attempt [**]
46 [**] [1:2003:14] SQL Worm propagation attempt [**]
24 [**] [1:1394:12] SHELLCODE x86 inc ecx NOOP [**]
8 [**] [1:1882:14] ATTACK-RESPONSES id check returned userid [**]
6 [**] [1:408:5] ICMP Echo Reply [**]
5 [**] [1:2129:10] WEB-IIS nsislog.dll access [**]
4 [**] [1:1243:20] WEB-IIS ISAPI .ida attempt [**]
1 [**] [1:542:14] CHAT IRC nick change [**]
1 [**] [1:498:7] ATTACK-RESPONSES id check returned root [**]
1 [**] [1:1887:5] MISC OpenSSL-Worm traffic [**]
```

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

Now, this honeypot is a Red Hat Linux box that has been compromised. When looking at the alerts generated by Snort, we need to eliminate those we deem informational or not pertinent to our environment – in this case, the honeynet.

ICMP messages can all be ignored since they appear to be informational. While they might be of interest to trouble shoot a network problem, or even have been an artifact of the scanning, they did not cause the compromise.

NETBIOS SMB logon failure... again could be a symptom of scanning, but it is a logon failure so we ignore these since there is no indication of success. We aren't concerned about the SQL and WEB-IIS alerts since they are associated with services that were not running.

What remains is of interest. The shellcode alert should be investigated since it may be an indication of a successful buffer overflow. However, it could also be a false positive; you have to determine which it is. The "ATTACK-RESPONSES" alerts could be a sign of compromise. These alerts fired ostensibly as a result of someone doing the "id" command to discover what user they currently are. An attacker is likely to do this since once compromised, they do not know the user associated with the exploited service once they gain access to the box.

The CHAT IRC is interesting since there was no IRC service running on the initial honeypot. The IRC nick change is interesting as IRC could be a sign that a successful attacker is inside the network and is either trying to communicate or control his malware.

You should be very suspicious of any alert that indicates the presence of a user on the host initiating outbound activity. After all this is a honeypot and no traffic should be initiated from it.

Alert of Interest

IP: 61.61.123.123 -> DST Port: 443

[**] [1:1394:12] SHELLCODE x86 inc ecx NOOP [**]
[Classification: Executable Code was Detected] [Priority: 1]
09/08-06:18:38.332774 61.61.123.123:33571 -> 192.168.1.3:443
TCP TTL:37 TOS:0x0 ID:18386 IpLen:20 DgmLen:256 DF
AP Seq: 0xD3EDFACF Ack: 0x9CF29F43 Win: 0x1DCE TcpLen: 32
TCP Options (3) => NOP NOP TS: 19054307 21861434

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

Here we have some information to drill down on, an IP of 61.61.123.123 and a destination port of 443. We also note that this is a priority 1 alert, which is described as "Shellcode x86 inc ecx NOOP", most likely indicates that this might be part of a buffer overflow.

As we discussed several time buffer overflows are often padded with NOOP instructions to allow the attacker to be less precise about the value of the address in the return pointer. The NOOP allows the return pointer to land in the NOOP sled of repeated NOOP's, ensuring that an executable instruction is found at the address of the return pointer and "sledding" along until the actual shell code is reached.

The following section will go into more detail on how to analyze this type of alert and how to identify if it is a true buffer overflow or if there is another reason why the executable code was detected. It will also cover how to use tools like Immunity Debugger to analyze the memory dump and find the exploit.

Immunity Debugger is a debugger specifically designed for analyzing memory dumps and finding exploits. It has a graphical user interface that allows you to view memory, registers, and stack, as well as tools for analyzing assembly and disassembly.

Drill Down (1)

- Now that we have some information, drill down for details
 - Tcpdump and Wireshark for packets
 - Unix tools for searching log data
- Example

```
tcpdump -nr challenge.pcap -w 61.61.123.123.pcap  
'host 61.61.123.123'
```

Now that we have some information to go by, such as IP, destination port, and type of attack (x86 NOOP buffer overflow) we can drill down. In the Challenge case, we have pcap data containing the packets from the many attacks. We can examine the packet data using tcpdump, Wireshark, or even using ngrep, pads, chaosreader, and many other packet analysis tools.

As an example, we read in the pcap files using tcpdump, and filter out all the packets to and from host 61.61.123.123 and save those packets to a file called 61.61.123.123.pcap :

```
tcpdump -nr challenge.pcap -w 61.61.123.123.pcap 'host 61.61.123.123'
```

Drill Down (2)

Intrusion Detection In-Depth

Here in the screenshot we see a following TCP stream from Wireshark. The primary filter was "ip.addr == 61.61.123.123." When a stream was followed, we see in red, traffic from the attacker, and in blue from the server. In red, towards the top we see the padding used for this buffer overflow represented by AAAAAAA's, or hex 41's. Following is the building of a shell script, and the results of the shell script in blue.

We can see in blue that the attacker's script generated some errors, but ultimately got user access to the server (uid 48).

Useful Information:

The purpose of the software that the attacker(s) downloaded:

Before beginning, here is an explanation of some of the names of files/software you should see in the reconstructed sessions. The attacker's motives and attempts to start or manipulate these files will make more sense if you know what they are:

pt or p	→	local ptrace root exploit binary
punk.c	→	backdoor source (65510)
fsflush	→	backdoor binary (65510)
qmail	→	backdoor binary (65519)
bnc	→	IRC bouncer (32700)

syslog traffic contains activity from the system administrator of the honeynet as well as the attacker.

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

- The information on this slide may help you assess what is transpiring. You will see some software that the attacker(s) attempt to install. Also, if you examine syslog traffic be aware that this reflects activity of both attacker and system administrator or the honeynet. The system administrator had to perform some activity such as restarting the system, perhaps to prevent the attacker(s) from targeting hosts external to the honeynet.

Analysis Write-up

- When, what, where, who, and maybe even why
- Begin by describing the event
 - Source and destination
 - Services attacked
 - Timing, packet number (Wireshark)
 - Description of attack

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

Once you feel like you have a grasp on the situation, you can begin your analysis write-up. You should record notes along the way as you pursue the investigation – not only to help you understand and remember what you've analyzed, but also in preparation for your final report.

Your report should include when the incident happened, what you believe occurred, who – what IP addresses were involved and perhaps why they attacked and what made it possible for the attack to be successful. Details should include the IP addresses involved, the services that were attacked, and the time of the attack to include Wireshark packet numbers. Finally, to the best of your ability and knowledge, describe exactly what you believe occurred. The more details that you can supply, the better you support your case, especially if it ends up in prosecution.

Unix Power Tools for Power Analysis

- Methodology Overview
- Unix Power Tools for Power Analysis
- Challenge Introduction
- Challenge Exercise
- Final Thoughts

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

This page intentionally left blank.

Most Useful Tools to Use Today

- Network Tools

- Tcpdump, Snort, Wireshark, Chaosreader, ngrep
- Refer to the Analyst Toolkit section for reference information

- Unix Tools

- `cut`, `sort`, `uniq`, `grep`
- We will go through a primer on these tools now...

The most important tools to use today are split into two main categories: Network tools and Unix tools. In some cases, one tool will complement the other, as we will see in the coming slides.

cut

- Used to cut or extract a field from a line
- Uses "tab" as primary delimiter
- Use -d to specify any delimiter you want
- Use -f to specify field(s) to cut

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

You can read the man page for the exact description and command line switches for the cut command. Generally, all you need to know for the Challenge is that the cut command is used to extract a particular field from a line. It could be used, for instance, if you want extract only the source IP's from tcpdump output.

There are really only two pieces of information you need to provide the cut command for our purposes in the Challenge. The -f indicates which field number(s) to cut. By default the field separator of a tab is assumed to divide all fields. Most of the time, this is not the case. So, you need to use the -d command line switch to specify perhaps a space " " or for tcpdump, perhaps a colon ":".

cut Examples

- This example will cut field #3 using a delimiter of space " "

```
cut -f 3 -d " "
```

- This example will cut fields 1 through 4 using a delimiter of period "."

```
cut -f 1-4 -d "."
```

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

These are useful examples of using the cut command. Say we have pcap file output and you wanted to extract the source IP field. Checking below, we see the source IP field is in field # 3 of the tcpdump command output if we used a space as our delimiter.

```
03:12:51.220891 IP 200.184.43.197.2776 > 192.168.1.3.443: . ack 3737596700 win 5840  
<nop,nop,timestamp 184801091 20672764>
```

You can see here using cut to extract the 3rd field, we get the source IP and source port.

```
tcpdump -nr challenge.pcap 'dst port 443' | cut -f 3 -d " "  
61.61.123.33587
```

We can now pipe "|" the output of the one command to another, to extract just the IP address and chop off the port.

```
tcpdump -nrr challenge.pcap 'dst port 443' | cut -f 3 -d " "| cut -f 1-4 -d ":"  
61.61.123.123
```

sort

- Used to “sort” the input, either alphabetically or numerically
- Often used in conjunction with other Unix tools
- Nice clean output
- sort options:
 - n is a numeric sort
 - r is reverse sort
 - u show only unique lines

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

Sort is used to organize output either alphabetically or numerically. Some of its most useful options are:

- n: this will sort the data numerically
- r: sort the output in reverse order
- u: sort the data and only show each unique line

Sort is a great utility; however, it is typically used in conjunction with other Unix commands. It lends itself to taking data from other tools, to well... sort out the output for you. Here is an example showing some of the earlier tools:

```
tcpdump -nr challenge.pcap 'dst port 443' | cut -f 3 -d " " | cut -f 1-4 -d "." | sort -u
```

```
200.184.43.197  
203.65.197.131  
61.61.123.123
```

uniq

- Show unique line entries
- Most implementations require sorting first
- Useful to find out how many of some form of data appear
- -c option will count how many unique line entries

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

The Unix tool uniq is very useful, again in conjunction with other text processing tools (such as sort, grep, cut, etc). Most implementations of uniq require sorting to take place prior to identifying unique line entries. Use the -c option to count how many of each entry appear in your data. Let's embellish our previous example to show us how many packets each IP sent to port 443:

```
tcpdump -nr challenge.pcap 'dst port 443' | cut -f 3 -d " " | cut -f 1-4 -d "." | sort | uniq -c | sort -rn  
  
989 200.184.43.197  
564 61.61.123.123  
4 203.65.197.131
```

We can see above that 1 IP sent 989 packets, another 564, and another only 4.

grep

- Quintessential Unix tool
- Regular expression search utility
- Searches for the pattern in each line of data provided
- Use to filter data
- Options: `-v` (skip if matches pattern); `-i` (case insensitive)

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

Grep is to many one of the core “can’t live without” tools that are part of every Unix distro. Grep is a search tool. It can be used to look for patterns, heck, even the absence of a pattern. One of the most powerful features of grep is that it can use regular expressions in its searches. Two useful options are: `-v` to skip a pattern:

```
grep -v 'hacker' fileinput
```

The above command would print every line in the file that did not contain the string: hacker

Another useful option is the `-i` switch. This turns off case sensitivity for grep.

```
grep -i 'hacker' fileinput
```

This would match “hAcKEr” for example.

Piping Everything Together

- As you have seen with many of our examples, these tools were made to be used together

```
command 1 | command 2 | command 3
```

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

The name of the game with Unix command line text processing tools is piping one tool to another to get the desired output.

Challenge Introduction

- Methodology Overview
- Unix Power Tools for Power Analysis
- Challenge Introduction
- Challenge
- Final Thoughts

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

This page intentionally left blank.

"The Challenge" SANS 503 Capstone Exercise

Honeynet gets compromised
multiple times by multiple
attackers

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

In this capstone exercise, we will put together the lessons we have learned during the course. This gives you an opportunity to explore many of the tools and techniques that you have learned.

Introduction

- Honeypot background
 - A new Red Hat Linux system is brought online
 - It is running a default installation and a standard set of services
 - The system is part of a GEN II Honeynet deployment
- Within a short time, the system is compromised
- Now we get to figure out what happened!

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

A good friend and fellow SANS Instructor, Jess Garcia, deployed a honeypot using a Red Hat Linux system. The system was set up as a Gen II Honeynet, using tcpdump to capture all the packets to and from the honeypot. You have a copy of these packets on your VM system under: /home/sans/Exercises/Day6/challenge.pcap.

The Red Hat system installed was current at the time, with a default set of services installed. You will see that these services get probed and attacked. One or more services get compromised. Your mission, should you choose to accept it, is to analyze the traffic and determine how the system got compromised, by which IP's, and any other information you can find out.

Objective: Analyze the Compromise

- Analyze the traffic
- Identify the attacks
- Look for false positives and false negatives
- Dissect the compromises
- Characterize the attackers
- Determine the extent of the compromise
- Correlate network and system information
- Learn to use the right tool for the right purpose
- Use the tools from the Analyst Tool section

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

This is a basic roadmap for analyzing the compromise. This is the same set of tasks that you will perform when you do your own investigations.

Challenge Exercise (1)

- Methodology Overview
- Unix Power Tools for Power Analysis
- Challenge Introduction
- Challenge Exercise
- Final Thoughts

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

This page intentionally left blank.

Challenge Exercise (2)

- Split up into teams
 - Recommended 4-8 people
- Use many different tools and techniques – don't get stuck
- Ask questions
- Break every 30 minutes for Situation Report
- Have fun!

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

In the classroom, we'll divide into teams. Remember if you do not get the answer you wanted from one tool, try another. We learned about many tools and most can help you in some fashion or other with your analysis today. If you are at home, make sure you take a break too every so often. When you return you can review what you've learned in the past half hour or so. Make sure you record notes as you proceed.

Suggested Path

- Run Snort
- Drill down the alerts – tcpdump, Unix tools, Wireshark
- Analyze suspected traffic in Wireshark
- Look at chaosreader for clues
- Use ngrep to look for suspicious keywords

The proposed path is to begin with Snort to find the indicators that will guide your analysis. Once you find alerts that you deem worth of pursuing, use other tools to discover details surrounding the alerts. As you know by now, Wireshark is an especially good tool for examining TCP streams of activity. Chaosreader can also help give you a summary of what transpired, while ngrep can help you look at the pcap for keywords that you use to find suspicious activity.

Alternate Path

- If you would prefer a more guided approach, follow the Day 6 Course Exercises in your workbook
- Gently guides you through the process if you feel lost
- Answers are available if you need more assistance

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

Another approach that you may elect to take is follow the steps and guidance found in your Day 6 Course Exercises in the workbook. We understand that students have a wide range of experience from those who are new to the field to those who feel quite comfortable digging in to a challenge without assistance.

If you feel stuck or do not feel you know how to proceed – don't worry. The instructor can give you some assistance if you would like. Alternatively, the Day 6 Course Exercises are supplied to assist you in discovery. Also, there are answers that follow if you'd like to check to see if you are on the right track. Don't panic; help is on the way!

Schedule

- Break up into groups
- Analyze & prepare write-up
- At 2:30PM we will share our results with our peers
- Following the presentations:
 - Instructor analysis of honeypot
 - Wrap-up

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

Our in-class schedule begins with dividing into groups. You are expected to analyze the Challenge data and prepare a write-up simulating a report that you would turn in for a real investigations. We'll convene at 2:30 PM to share our findings with our peers. We'll end the day with a discussion of the instructor analysis of the attacks of the honeypot and there will be a final wrap-up.

Those of you at home may also find value in preparing a write-up. It's not quite the same as sharing with others, but it will give you practice for any future investigations you do.

Final Thoughts

- Methodology Overview
- Unix Power Tools for Power Analysis
- Challenge Introduction
- Challenge Exercise
- Final Thoughts

© SANS,
All Rights Reserved

Intrusion Detection In-Depth

This page intentionally left blank.

SEC503 Final Thoughts

In conclusion ...

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

Let's wrap up this week with some final thoughts.

The first thing I would like to say is that I am very excited about the new version of the course. It has been a long time coming and I am very happy to see it finally released. I think it is a great improvement over the previous version and I hope it will continue to evolve and improve over time.

I have found many similarities between myself and my students, particularly those who are new to the field. We all seem to have a similar level of interest in security and a desire to learn more. I think this is great because it allows us to work together and share our knowledge and experiences. I also appreciate the emphasis on practical skills and hands-on experience. This is something that I believe is essential for success in the field of cybersecurity. I hope that the new version of the course will continue to focus on these important aspects and provide valuable learning opportunities for everyone involved.

Overall, I am very excited about the new version of the course. I believe it will be a valuable resource for anyone interested in learning more about cybersecurity. I hope that it will help to inspire and motivate others to pursue a career in this field. I am looking forward to seeing what the future holds for the field of cybersecurity and I am grateful for the opportunity to be a part of it.

Finally, I would like to thank all of the students who have taken the course so far. Your enthusiasm and dedication to learning is truly inspiring. I hope that you will continue to pursue your goals and dreams in the field of cybersecurity. I am confident that you will succeed and make a positive impact on the world. Thank you for taking the time to read this message. I hope you found it informative and useful.

Course Summary (1)

- You have many open source tools at your disposal
- You can't detect an intrusion without an appropriate signature or protocol decode
- Rule customization is key to success
- Use/correlate log files as a valuable source of data

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

Let's summarize some of the most important points we'd like you to remember. We've covered many open source tools – some are more general in their capabilities, such as Wireshark and tcpdump. Others perform very specific functions such as p0f for passive operating system identification. There is probably an open source tool that can do most anything you need to do. You just need to identify the proper tool for the given task.

While it may seem absolutely obvious to you in theory, you cannot detect an intrusion without some kind of signature or rule, protocol decode, or perhaps anomaly detection. Remember, just because you get a boatload of signatures from either a commercial tool or from open source like Snort – these signatures are not customized for your site. You have to tune out the false positives in order to get relevant information. And just because you've customized your signatures or configuration options doesn't mean that there will be a signature for every new exploit that shows up. Many argue that anomaly detection and protocol decode are better for discovering zero day exploits and studies show some truth to that since new exploits often violate expected behavior. The bottom line is don't become complacent just because you have an IDS/IPS with a bunch of signatures. That doesn't necessarily mean that you are going to see every intrusion or attack.

A lone IDS/IPS is better than nothing at all, but that isn't the best possible solution. There are many other sources of data in a well-defended network to use for correlation. One of the most important is the firewall or packet-filtering logs. Make sure you examine those and correlate them with your IDS/IPS output. They help you get a sense of what is the most harmful traffic and traffic that was shunned from the site. There are other sources of useful traffic analysis such as some kind of central logger that permit you to gain a far better perspective than using a single IDS or IPS.

And while we're talking about logs - don't forget about logs on individual hosts such as web servers or individual host syslog output. There are plenty of tools that can help you analyze host traffic such as personal firewalls, HIPS, TCPWrappers, to name a few. In this new world of detection, we correlate everything including vulnerability scans with traffic seen by all of our other tools. This will determine whether any suspicious traffic is aimed at a vulnerable target.

Course Summary (2)

- Ultimately your search for, and success in, detection will be based on:
 - What is the trigger/catalyst for the intrusion?
 - Has the proper data been captured and retained?
 - Do you have the proper tools to analyze the data?
- Just when you think you have a clue, everything changes

© SANS.
All Rights Reserved

Intrusion Detection In-Depth

As you've just experienced, trying to discover a compromise and all the surrounding details can be a very tricky ordeal.

Your best chance for discovery and analysis are dependent on many facets. First, obviously, you need to see some indication of the intrusion. In terms of pride, it is best that you or someone in your network or security team finds it, rather than being informed by another site that thinks you are attacking them.

Capturing data associated with the incident is key. First, you must have some kind of sensor in a location on the network that sees the data. It is possible that if a network is large and the traffic is voluminous, the security budget may not allow you to deploy as many sensors as you'd like, leaving you blind to certain traffic. Speaking of volumes of data, it may not be possible to retain as much as you'd like – especially full packet captures. Many sites retain historic data for long periods of time; however, this is more likely to be summarized data, such as flows.

Next, you need tools to scrutinize data. You've seen just a handful this week – many are open source. The more tools you have and the more varied they are in capabilities and purpose, the better. As the old expression goes "there are many ways to skin a cat" (but really, who skins cats anyway?).

Finally, there is another challenge and that is change. If you've been involved with security work for even a little while, you know that there is rapid change in attacks, attacker's methods, tools, and new protocols and devices. However, with the proper tools and knowledge, it becomes manageable challenge.

Good luck in your analysis work and defense of your network!

ABOUT SANS

SANS is the most trusted and by far the largest source for information security training and certification in the world. It also develops, maintains, and makes available at no cost the largest collection of research documents about various aspects of information security, and it operates the Internet's early warning system – the Internet Storm Center. The SANS (SysAdmin, Audit, Network, Security) Institute was established in 1989 as a cooperative research and education organization. Its programs now reach more than 165,000 security professionals around the world. A range of individuals from auditors and network administrators to chief information security officers are sharing the lessons they learn and are jointly finding solutions to the challenges they face. At the heart of SANS are the many security

practitioners in varied global organizations from corporations to universities working together to help the entire information security community. SANS provides intensive, immersion training designed to help you and your staff master the practical steps necessary for defending systems and networks against the most dangerous threats – the ones being actively exploited. This training is full of important and immediately useful techniques that you can put to work as soon as you return to your office. Courses were developed through a consensus process involving hundreds of administrators, security managers, and information security professionals, and they address both security fundamentals and awareness and the in-depth technical aspects of the most crucial areas of IT security. www.sans.org

IN-DEPTH EDUCATION AND CERTIFICATION

During the past year, more than 17,000 security, networking, and system administration professionals attended multi-day, in-depth training by the world's top security practitioners and teachers. Next year, SANS programs will educate thousands more security professionals in the US and internationally.

SANS Technology Institute (STI) is the premier skill-based accredited cybersecurity graduate school offering master's degree in information security. Our programs are hands-on and intensive, equipping students to be leaders in strengthening enterprise and global information security. Our students learn enterprise security strategies and techniques, and engage in real-world applied research, led by the top scholar-practitioners in the information security profession. Learn more about STI at www.sans.edu.

Global Information Assurance Certification (GIAC)

GIAC offers more than 27 specialized certifications in the areas of incident handling, forensics, leadership, security, penetration and audit. GIAC is ISO/ANSI/IEC 17024 accredited. The GIAC certification process validates the specific skills of security professionals with standards established on the highest benchmarks in the industry. Over 65,000 GIAC certifications have been granted with hundreds more in process. Find out more at www.giac.org.

SANS BREAKS THE NEWS

SANS NewsBites is a semi-weekly, high-level executive summary of the most important news articles that have been published on computer security during the last week. Each news item is very briefly summarized and includes a reference on the web for detailed information, if possible. www.sans.org/newsletters/newsbites

@RISK: The Consensus Security Alert is a weekly report summarizing the vulnerabilities that matter most and steps for protection. www.sans.org/newsletters/risk

Ouch! is the first consensus monthly security awareness report for end users. It shows what to look for and how to avoid phishing and other scams plus viruses and other malware using the latest attacks as examples. www.sans.org/newsletters/ouch

The Internet Storm Center (ISC) was created in 2001 following the successful detection, analysis, and widespread warning of the LiOn worm. Today, the ISC provides a free analysis and warning service to thousands of Internet users and organizations and is actively working with Internet Service Providers to fight back against the most malicious attackers. <http://isc.sans.org>

TRAINING WITHOUT TRAVEL

Nothing beats the experience of attending a live SANS training event with incomparable instructors and guest speakers, vendor solutions expos, and myriad networking opportunities. Sometimes though, travel costs and a week away from the office are just not feasible. When limited time and/or budget keeps you or your co-workers grounded, you can still get great SANS training close to home.

SANS OnSite Your Schedule! Lower Cost!

With SANS OnSite program you can bring a unique combination of high-quality and world-recognized instructors to train your professionals at your location and realize significant savings.

Six reasons to consider SANS OnSite:

1. Enjoy the same great certified SANS instructors and unparalleled courseware
2. Flexible scheduling – conduct the training when it is convenient for you
3. Focus on internal security issues during class and find solutions
4. Keep staff close to home
5. Realize significant savings on travel expenses
6. Enable dispersed workforce to interact with one another in one place

DoD or DoD contractors working to meet the stringent requirements of DoD-Directive 8570? SANS OnSite is the best way to help you achieve your training and certification objectives. www.sans.org/onsite

SANS OnDemand Online Training & Assessments – Anytime, Anywhere

When you want access to SANS' high-quality training 'anytime, anywhere', choose our advanced online delivery method! OnDemand is designed to provide a very convenient, comprehensive, and highly effective means for information security professionals to receive the same intensive, immersion training that SANS is famous for. Students will receive:

- Up to four months of access to online training
- Hard copy of course books
- Integrated lectures by SANS top-rated instructors
- Progress reports
- Access to our SANS Virtual Mentor
- Labs and hands-on exercises
- Assessments to reinforce your knowledge throughout the course

www.sans.org/ondemand

SANS vLive Live Virtual Training – Top SANS Instructors

SANS vLive allows you to attend SANS courses from the convenience of your home or office! Simply log in at the scheduled times and join your instructor and classmates in an interactive virtual classroom. Classes typically meet two evenings a week for five or six weeks. No other SANS training format gives you as much time with our top instructors.

www.sans.org/vlive

SANS Simulcast Live SANS Instruction in Multiple Locations!

Log in to a virtual classroom to see, hear, and participate in a class as it is being presented LIVE at a SANS event! Event Simulcasts are available for many classes offered at major SANS events. We can also offer private Custom Simulcasts – perfect for organizations that need to train distributed workforces with limited travel budgets. www.sans.org/simulcast

For group programs, please contact us at groupsales@sans.org