# Global Information Assurance Certification Paper

# Intruder Alert!
## The tools to piece the puzzle together

GIAC Security Essentials Certification (GSEC) Practical Assignment v 1.4

Chris Prickaerts
August 10, 2002

*Introduction*

It is Thursday, 3:45 in the afternoon, you get a call from one of the helpdesk employees. Calls seem to be pouring in, there seems to be something wrong with your company's website. "That's odd", you think: "the machine was running smoothly only a few hours ago, what might have happened?" You start up your web browser to check out t he website. You expect it not to show, but to your surprise the browser starts loading a page, but it is not the one your company's webmaster set up. The text displayed leaves no room for doubt about what might have happened, "You've b33n Own3d"...

So here it starts, at least that's the case for far too many administrators whose investigative work is made more difficult due to lack of evidence and pointers as to what has happened. Often simply because they had not yet defined their security policy. There a re many, perhaps too many, tools to 'get secure', just as there is an abundant supply of tools and techniques to circumvent them. Which tools you choose to deploy often depends on personal experiences, background and/or preference. "Many roads lead to Rome ", as a Dutch saying goes. My main focus will be on the relationship between logging and detectin g an attack, being able to understand what happened, estimating the scope of the damage and reacting appropriately.

I will elaborate on this issue keeping in mind the three security phases known to us: prevention, detection and response. I will follow these phases with the six stages of the PDCERF [1] as a guide to how logging should be deployed, how obtained information can be analyzed and what it can tell you. I won't discuss regular system monitoring too much. It provides valuable information on normal system/network behaviour needed to notice when something is wrong. Although equally important, I consider this to be part of any normal system maintenance. A lot of work will be done in the Prevention/ *Preparation* phase, since this is where you define what to log and how to log it. Further on I will focus on what this information can tell you. Finally, I will demonstrate how these instruments can severely influence the outcome of an intrusion.

**Prevention**

Preparation: Make sure your network is not wide open to anyone. Protect your users' data from unauthorized access and make sure you have those backups running. Write up a plan on how to respond to an incident and appoint the people to handle it when time comes. Make sure they know what to do, and in case they don't, whom they can consult.

There are some basic considerations when designing your logging environment[2]:

- the location of log files; on the system itself or on a remote logging host accessed via the network
- the expected size of log files
- the rate at which data is logged to the log files
- who needs access to the log files and what level of access they should have
- whether or not logging is to be encrypted
- how log files are to be backed up and recovered
- how long log files are to be retained

Since log files can provide insight into what happened during an intrusion, hackers tend to focus part of their attack on them. Covering their tracks by (sometimes partly) deleting them. That's why it is a good idea to set up a host to centrally collect log files. This host should be more secure and preferably be on a separate network. You should also have an idea of how much disk space log files use. You don't want your server crashing due to lack of disk space. On the other hand, disk space is inexpensive, so if you want to cut corners, do it elsewhere. Furthermore, who should be able to access the log files and how long should they be able to access them? Also, you do not want to find yourself destroying valuable data due to unclear policies regarding incident handling. Document the logging capabilities your site has deployed, where those files are sent to and who has access to them.

Do you have a firewall running to protect your network? If not, consider the option! It is beyond the scope of this document to discuss the pros and cons of having a firewall. But best security practices always have this one on the list[3]. Not only can it protect your network from harm, it can also provide good information on what's happening on your network. Let's say you have a firewall up and running, check to see if logs are retained, if only for a couple of days, if only for those packets being dropped. But if logging volume permits it, get it.

Now the question arises what should go into the logging. There's some basic traffic that should get your interest[4]. Incoming traffic pretending to be from your network is a clear indication of trouble. Have your router drop these packets and have them logged. Although the log won't point back to any perpetrator (since they pretend to be coming from your network), it will still indicate that someone is trying something at that specific time/day. It might be an indication of more to come. Same goes for outgoing traffic pretending to be from another network than yours. This might indicate one of your users took up the art of NMAP after reading a Hacking Exposed book. Or worse, one of your hosts has been compromised and is being used for reconnaissance work or other forms of hacker activities. Above all, don't under-estimate the amount and value of information that can come from logging rule compliance at a router/firewall level.

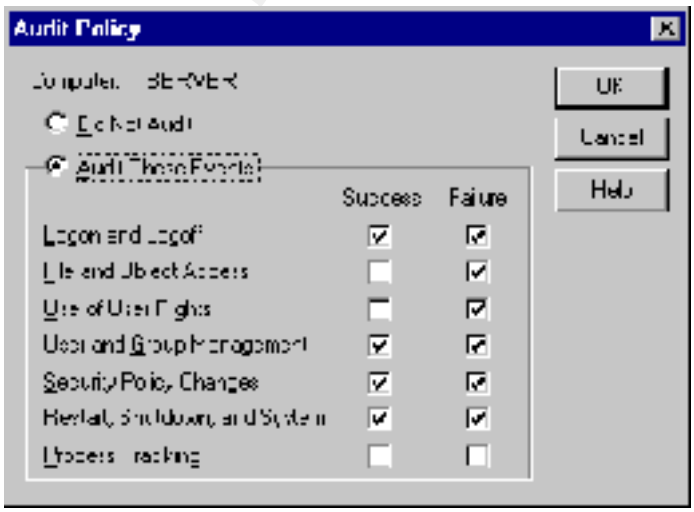
Things your router/firewall logs can tell you:

- when the attack started
- where it was coming from
- if other suspicious connections were made from that ad dress(space)
- if other suspicious connections were made with the victim system during the attack

It is no secret that the amount of vulnerabilities is still growing. Cert reported 1,090 vulnerabilities in the year 2000, 2,437 in 2001 and 2,148 in the first two quarters of 2002. In the five preceding years (1995 -1999) the added total lies around 1500[5]. This is one more reason to have your systems patched at all times. Add patching pro cedures to your security policy, audit your systems and where necessary apply relevant updates. Subscribe to your vendors security related mailing list (if they have one), to be sure you're informed quickly of any newly discovered vulnerabilities. Also che ck out CERT's list at www.cert.org

Out of all incidents reported to Cert most systems were misconfigured or unpatched[6]. Leaving no doubt as to whether patching is important. Linda McCarthy describes many incidents in h er book "Intranet Security, stories from the trenches" where her help as a security expert/incident handler was called upon[7]. In many, if not most of her stories, out -of-the-box installations provided the main opportunity for hackers, expanding their influ ence from there. And what's even more distressing is that often logging is not turned on by default, or only for the most basic information.

What should you log and what should you leave? There's not really a right answer to this[8]. It not only depends on the level of security you would like to live up to, it also depends on the type of services you are offering (web, ftp, DNS, etc) the potential amount of events that will be logged and the tools used to get information out of them. Above all it is very ha rd to predict what information is relevant when investigating a specific attack. There are however some guidelines on what to log in (for example) a MS Windows NT environment. Depending on what role the server has you might want to tune them down or up, bu t the basic setup should look like this [9]:



One major drawback with current MS Windows NT logging is that it still does not record the IP address that belongs to the computer from which the event was triggered. Although an IP -address can be forged, a net bios name can be faked more easily, without disrupting network functionality.

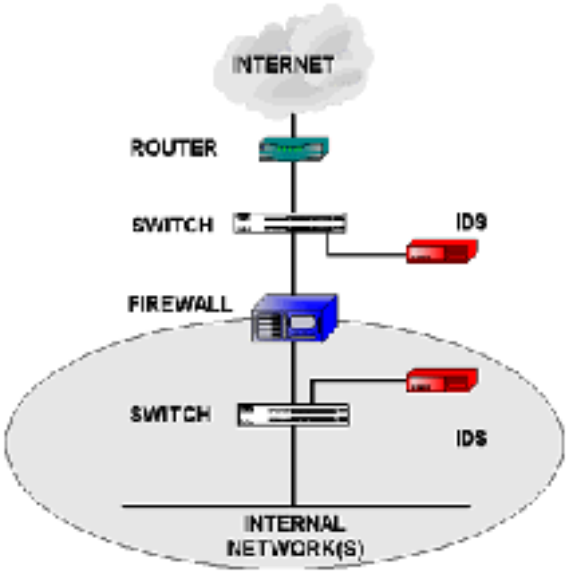
The idea is not to get swamped with information you don't want, but to still have the events logged that are important in case of an intrusion. You never know in advance what in formation holds the key to understanding what happened during an intrusion. Consider you're just logging failed login attempts. What if someone starts pounding your machine with the administrator account? Bad login attempts stick out like a fly, especially when they come in tens or hundreds at odd hours. This will show whether someone's trying to guess the admin password. If you are not logging successful attempts however you won't really know if he succeeded (unless you are really confident about the chose n password). So you decide to start logging successful login attempts. On a busy day this might flood your log files pretty fast, making it harder to spot the odd one out.

**Detection**

Detection: Set up (intrusion) detection mechanisms so that you are told about a possible intrusion, check your logs, update your virus scanners, and get to know the signs. You have to have an idea what normal system and network performance is to notice when something is out of the ordinary. Run regular checks on system perfor mance, log running processes, check bandwidth usage, and compare that with archived information.

How do you get your defenses up easily and at the same time improve your situational awareness? And how high are you going to set your ambition level. Do you want to be paged about every port scan that hits your network, or only when the web server stops responding? This is very important, because the first might not only make the phone company a lot of money and leave your pager flooded, it is also harder to set up and tune correctly. At first keep it simple. You can always expand and improve your defenses at a later stage. But make sure the groundwork is done right.

A network intrusion detection system is a device that monitors all (or parts) of the traffic on a network and looks for any suspicious activity [10]. It does this by checking every packet it sees and comparing the content with a set of rules. These rules can be anything, from checking if the packet is part of a telnet session to looking for a specific string in a HTML request. You could even use an IDS to see if your employees are viewing less work -related adult sites, it all depends on how the rules are defined. The primary task of an IDS, of course, is to check for suspicious network traffic, and not to snoop on employees. When an IDS detects traffic that matches one of its rules, it can do several things. It can log information about the packet, where it came from and where it was going, it can store the content, it can trigger an alert or send out a notification. Form and method differ from product to product, but the basic principles are the same.




Often a network intrusion detection system is set up to listen in on the network in front of the firewall and also behind, as shown in this picture. That way it can detect and identify attacks hitting the firewall and see which ones get through. (picture courtesy of Scorpionpoint Security, www.scorpionpoint.com)

One of the drawbacks of a network intrusion detection system is that it can detect a hacking attempt, but in some cases the created logs won't show if it was successful. Let's suppose someone is trying out the Unicode attack on one of your web servers. Your newly setup Snort system detects this attempt, since it spotted specific pa tterns in packets passing the network. However, it won't know if your web server was patched and withstood the attack. It won't even know if your web server is vulnerable to this type of attack. To deal with this gap in your overall view you can setup host -based intrusion detection software.

There are several different groups of host -based intrusion detection products. Programs like the well -known Black ICE that serve as a firewall/IDS combination. Although not perceived as such by many, a virus scanner is also a form of an intrusion detection system, keeping out trojans and worms. Their log files harbor great information on how the attack took place and might tell you if it was successful.

Besides checking incoming traffic it is also possible to track ch anges on system files. Programs like Tripwire ( http://www.tripwire.com ) can accomplish this. When you have a freshly installed system, let Tripwire run a checksum. It will keep a database of installed files and then cryptographically sign the result. Regular checks are compared to the baseline result, detecting any changes made to the system. If a change has been detected, an alert can be triggered, an event log added or an admin paged. You can also frequently check the services running to see if any service out of the ordinary starts running, this might be a sign of a worm, keystroke logger, password cracker or any other kind of rogue software.

Log who accesses the machine, via the network or interactively. Employ file checksum software, so any unwanted alterations to important system files (or additions) can be tracked. The most important thing is that you have data to

dig into when your investigation starts. Nothing worse than to discover logging was not activated on the machine hacked into.…

**Response**

Containment: Assess the scope, impact and damage of the attack and take actions to stop an intruder's access to compromised systems (thereby limiting the extent of an intrusion) and prevent an intruder from causin g further damage. Assess the possible magnitude of the break -in, how many systems are compromised, are privileges elevated, has any data been destroyed? Is the intruder still roaming your network? If so, do you let him continue to see what he is doing and on what systems he has been? You might increase the level of monitoring to get a better picture. Or do you disconnect him and start your investigation? Then decide if you want to shut down the compromised host or take it off the network.

Make these decisions based on the information and experience at hand to limit the extent of an attack, thereby limiting the amount of damage being done by an attack, be it physical, monetary or reputation. Here the information you gathered at different levels (perimeter, network, host) will be used. Were more connections from the same attacking host logged at router level? Did the network intrusion detection system pick up the attack? Did other hosts have any connections to the attacked host? Were there connections made fr om the attacked host at the time of the attack?

To get an idea of the scope and damage of an attack you need to find out what happened. In case of an incident you want to be able to get an idea of the attack vector used. You can try to find out by looking through the loggings. All you need is a date/time to get your investigation started , the tools to scavenge the log files for clues and the ability to interpret them.

This is where event log tools come into the picture. They take care of processing all the information gathered and presenting it in a readable form, sometimes leaving out unwanted entries. Every platform has its own set of tools, in different flavours. Unix has Logwatcher, Swatch and Windows has several resource kit utilities (dumpel.exe, ev entquery.pl), ntlast (by foundstone) languard, ELM, etc. Which tool to employ depends on several factors: How many systems are involved, how much data there is to analyze, how much time you are willing to spend going through it, and how much money you can spend.

Eradication: Repair the damage; get rid of attack tools (root kits). And most importantly, eliminate the cause of the intrusion. Obviously, you should be able to ascertain the cause to be able to succeed at this. Comparing normal file-system inform ation with the attacked host should reveal the changes made to the system. Look for strange processes running, files out of the ordinary, compare the system with documentation made at installation. Decide if the system should be rebuilt entirely, or only m issing data replaced.

Recovery: Get your systems back up and running doing the job they're supposed to at the level they were expected to before the attack. Be sure you fixed the holes that made the attack possible. Review the changes made to your systems and if necessary update you documentation.

Follow-up: Get your incident response team together and review what happened. Were you able to respond adequately and fast enough? Did the procedures help in streamlining the activities? Did the logging mechanisms work properly? Were you able to use all the information that was gathered? This is a very important phase. It should be used to review if your procedures need any changing. Discuss points of improvement and if you were able to respond adequately based on the information obtained from the various log files/instances.

*The case presented*

So here it goes, pointing out where all these mentioned tools and procedures come to light and help your life as an incident handler (that's what you become the moment you respond to that defacement call). It is important to restate the goal set out in this paper. Collect as much information as you can about an attack that can help you when handling an intrusion (note that I use 'when', not 'if'). This will help you to prevent an attack, detect one as it occurs and react to it if your defenses fail. I'll depict the scenario from the beginning again, first the case in which no defensive measures were taken or implemented badly. Then what could have been if a little more time had been put into taking security a bit more seriously. And finally if you take your job as a security-minded administrator seriously and are given the time and resources to implement it.

*"Have you seen our website?"*

You got the call, something has happened to your website, you take a look and to your dismay you don't see the homepage you're used to. It is obvious what has happened, it must be obvious to the rest of the online world too… 'You've b33n Own3d". You quickly grab that cup of coffee, and start looking for the backup you made just yesterday, wondering what might be the best course of action at this moment. First things first you tell yourself, get that website back up in its old glory and worry about things later. Fifteen minutes later the website is restored, and you put on a smile. They'll have to do better than this to get you sweating you say to yourself. But what if they did? You decide to check the logs, if only for routine's sake. Your firewall log doesn't show much, but that might be because you're not logging rule compliance. Your IIS NT server doesn't show much either. It's no use to browse the IIS logs. Since your website has a lot of traffic you decided it would only eat up valuable disk space and turned it of. You don't really have time to wonder how this could have happened, because the phone's ringing again "why you haven't fixed the problem yet, the website is still defaced….."

*"There's a problem with our company website, but I'm already working on it."*

You got the early warnings from your network IDS, apparently someone was looking for web servers within your network. After the first series of port probes the attack took off. HTML requests were pouring in for your web server. Snort identified them as a Unicode attack. You're not f amiliar with that one, so you get on the Internet and do some research. It doesn't take long to find some good descriptions, tools and security advisories. You have a look at your website and indeed the web page has been defaced, rather crudely. You check the IIS logs and you do a quick search for %c0% and sure enough, around the time the site was defaced the web server processed these request. It seems you forgot to run that patch mentioned a couple of weeks ago. Ah well, anyone can make a mistake. You inf orm the helpdesk that the web server will be down for half an hour. You restore the website from backup tapes and run the freshly downloaded required patches. After rebooting the server you bring it back online and check to see if anything else seems strange about it. What you can tell from this kind of attack is that it is mostly a website harassment kind of attack, but it's better to be safe then sorry.
You check the IDS logs to see from what IP address these requests were made, and if they are the same y ou saw in the IIS logs. They are. You make a note of the address and go through the rest of the IIS logs to see if this computer has visited your site earlier. You also check your proxy firewall logs to see if he tried to pass the DMZ. All in all an exciti ng afternoon, you write up a report and feel glad you set up that IDS. It paid off already.

*They tried to enter, but were left cold at the front door, beaten before the battle.*

You saw this one coming miles away. Not only because your network IDS started informing you of increased probe activity looking for web servers inside your network, the moment you saw the Unicode flag being raised you grinned. Sure, you have a website in the DMZ visible to the Internet, but it was patched weeks ago against this ki nd of attack. The attack was withstood before it even took place. Hell, why not have a little fun with it. It might be a script kidd ie, but then again it might be someone really wanting to harm your company. Industrial espionage and sabotage are not a thin g of the past in the Internet realm. You start up your decoy web server and wait until the attacker moves in on this easy prey. He quickly finds the machine and does his magic. You make sure all loggings are recording this. After the initial defacement the attacker goes on, this time trying to get a remote shell using a known buffer overflow attack. "Must be using Netcat" you think, "that's what I'd do". Sure enough, he gets in and starts uploading his root kit. Not much later he's trying to set up shop at your freshly opened honey pot. Time to close the trap, mustn't harbour any happy hacker. You disconnect the user, take the web server offline, adjust your router so that the address is rerouted to digital outer space and start collecting the evidence. Just too easy….

### *Conclusion*

Although the above scenarios might seem very extreme, they are not. The Internet community is shaping up, sure. People are spending more and more time in setting up security. Sometimes because they have to. Sometimes

because they want to. And sometimes because it is what they've always done. Fact is that the number of incidents is still on the rise. Too many people are not prepared, lack the funds, are not aware or just don't care.

One needs to realise that to be able to handle an intrusion well, one has to be able to find out what happened. The secret to success lies in doing your homework. Knowing the default logging capabilities and how to extend them. This way you are able to choose how to implement your defenses more effectively. It is all about the relationship between several layers of defense, the information you can obtain at that level and how the tools mentioned can help ascertain what really happened during the intrusion. Who tried doing what to whom and how did they do it? Putting the pieces together does not have to be difficult. The clarity of the picture just depends on the number of pieces you have at your disposal. Collect those pieces at every defense layer you install. When the day comes you will be glad you did.

**References:**


[1] Schultz, Eugene E & Shumway, Russell. "Incident Response, a strategic Guide to Handling System and Network Security Breaches". Indianapolis, New Riders Publishing, 2001. p 45 -71

[2] "Configure firewall logging and alert mechanisms" http://www.cert.org/security -improvement/practices/p059.html (10 august 2002)

[3] "Seven Simple Computer Security Tips for Small Business and Home Computer Users" taken from National Inf rastructure Protection Centre website. http://www.nipc.gov/warnings/computertips.htm (10 august 2002)

[4] "Improving Security on Cisco Routers" http://www.cisco.com/warp/public/707/21.html#logging (10 august 2002)

[5] "CERT/CC Statistics 1988 -2002" http://www.cert.org/stats/cert_stats.html#vulnerabilit_ies (10 august 2002)

[6] "Cert/CC Overview, Incident and Vulnerability Trends" http://www.cert.org/present/cert -overview-trends/module -4.pdf (10 august 2002)

[7] McCarthy, Linda . "Intranet Security, stories from the trenches". Sun Microsystems Press, Prentice Hall, 1998.

[8] Allen, Julia "Identify data that characterize systems and aid in detecting signs of suspicious behavior" http://www.cert.org/security -improvement/practices/p091.html (10 august 2002) (also in print, Addison -Wesley 2001)

[9] "Effective Security Monitoring", Chapter 4 from Microsoft Windows NT 4.0 Security, Audit, and Control, published by Microsoft Press. http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/prodtech/network/ch04.asp (10 august 20 02)

[10] Mell, Peter & Bace, Rebecca. "Nist Special publication on Intrusion Detection" http://www.snort.org/docs/nist -ids.pdf (20 July 2002)

Laing, Brian. "Implementing a network based intrusion detec tion system URL: http://www.snort.org/docs/iss -placement.pdf (10 august 2002)

Rosato, Rick. "Best Practices for Applying Service Packs, Hotfixes and Security Patches". http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bestprac/bpsp.asp (10 august 2002)

Scambray, Joel & McClure, Stuart. "Hacking Exposed, window s 2000 edition". Berkeley, Osborne / McGrawHill, 2001.

Last Updated: July 24th, 2017

# Upcoming Training


| | | | |
|---|---|---|---|
| SANS Prague 2017 | Prague, Czech Republic | Aug 07, 2017 - Aug 12, 2017 | Live Event |
| SANS Boston 2017 | Boston, MA | Aug 07, 2017 - Aug 12, 2017 | Live Event |
| SANS New York City 2017 | New York City, NY | Aug 14, 2017 - Aug 19, 2017 | Live Event |
| SANS Salt Lake City 2017 | Salt Lake City, UT | Aug 14, 2017 - Aug 19, 2017 | Live Event |
| Community SANS Omaha SEC401* | Omaha, NE | Aug 14, 2017 - Aug 19, 2017 | Community SANS |
| Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style | Virginia Beach, VA | Aug 21, 2017 - Aug 26, 2017 | vLive |
| SANS Chicago 2017 | Chicago, IL | Aug 21, 2017 - Aug 26, 2017 | Live Event |
| SANS Virginia Beach 2017 | Virginia Beach, VA | Aug 21, 2017 - Sep 01, 2017 | Live Event |
| SANS Adelaide 2017 | Adelaide, Australia | Aug 21, 2017 - Aug 26, 2017 | Live Event |
| Community SANS Pasadena SEC401 @ NASA | Pasadena, CA | Aug 23, 2017 - Aug 30, 2017 | Community SANS |
| Mentor Session - SEC401 | Minneapolis, MN | Aug 29, 2017 - Oct 10, 2017 | Mentor |
| SANS San Francisco Fall 2017 | San Francisco, CA | Sep 05, 2017 - Sep 10, 2017 | Live Event |
| SANS Tampa - Clearwater 2017 | Clearwater, FL | Sep 05, 2017 - Sep 10, 2017 | Live Event |
| Mentor Session - SEC401 | Edmonton, AB | Sep 06, 2017 - Oct 18, 2017 | Mentor |
| SANS Network Security 2017 | Las Vegas, NV | Sep 10, 2017 - Sep 17, 2017 | Live Event |
| Mentor Session - SEC401 | Ventura, CA | Sep 11, 2017 - Oct 12, 2017 | Mentor |
| Community SANS Albany SEC401 | Albany, NY | Sep 11, 2017 - Sep 16, 2017 | Community SANS |
| Community SANS Columbia SEC401 | Columbia, MD | Sep 18, 2017 - Sep 23, 2017 | Community SANS |
| Community SANS Dallas SEC401 | Dallas, TX | Sep 18, 2017 - Sep 23, 2017 | Community SANS |
| SANS London September 2017 | London, United Kingdom | Sep 25, 2017 - Sep 30, 2017 | Live Event |
| SANS Baltimore Fall 2017 | Baltimore, MD | Sep 25, 2017 - Sep 30, 2017 | Live Event |
| SANS Copenhagen 2017 | Copenhagen, Denmark | Sep 25, 2017 - Sep 30, 2017 | Live Event |
| Community SANS Boise SEC401 | Boise, ID | Sep 25, 2017 - Sep 30, 2017 | Community SANS |
| Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style | Baltimore, MD | Sep 25, 2017 - Sep 30, 2017 | vLive |
| Community SANS New York SEC401 | New York, NY | Sep 25, 2017 - Sep 30, 2017 | Community SANS |
| Rocky Mountain Fall 2017 | Denver, CO | Sep 25, 2017 - Sep 30, 2017 | Live Event |
| Community SANS Sacramento SEC401 | Sacramento, CA | Oct 02, 2017 - Oct 07, 2017 | Community SANS |
| SANS DFIR Prague 2017 | Prague, Czech Republic | Oct 02, 2017 - Oct 08, 2017 | Live Event |
| Community SANS Charleston SEC401 | Charleston, SC | Oct 02, 2017 - Oct 07, 2017 | Community SANS |
| Mentor Session - SEC401 | Arlington, VA | Oct 04, 2017 - Nov 15, 2017 | Mentor |
| SANS Phoenix-Mesa 2017 | Mesa, AZ | Oct 09, 2017 - Oct 14, 2017 | Live Event |