Global Information Assurance Certification Paper

Securing the Telephony Server

A Case Study

Tim Hoskins
GIAC Security Essentials Certification (GSEC)
May 5, 2004
Practical Assignment Version 1.4b Option 2

Securing the Telephony Server

Abstract

Securing the telephony server is a real-world solution in which I will be evaluating the security of an existing telephony system prior to its integration into customer networks. The telephony system discussed in this paper is a standalone system that typically has no need for network integration. However, as the product has evolved, so has this need. Currently these systems reside on site at each customer location and are only connected via an existing telephony interface. Changes in the market have spawned enhancements in the product line that will require network integration to facilitate electronic transactions to third parties.

A majority of the security vulnerabilities have already been identified earlier in the project through peer and third-party review of the system. These identified vulnerabilities have been essentially handed to me as "known issues". The purpose of this paper is to chronicle my journey through the first phase of my proposed security plan to correct these issues. This final plan must resolve the previously identified vulnerabilities prior to network integration of customer systems. I will share with you my pain, joy, and frustration as I work toward implementing my security strategy that in the end will provide a robust security framework.

The Test System

To start the process I built a working system to replicate the systems currently installed at customer sites. For ease of reference we will nickname this system "Hackme" and I will refer to it this way for the remainder of the paper. The Hackme system consists of the following components:

Hardware

- Dell PowerEdge 2600 server (single 2.3GHZ Zeon with 1GB memory)
- 20GB primary volume housing OS and 115GB secondary data drive
- Intel Dialogic D120JCT-LSU PCI Telephony Board

Software

- Microsoft SQL Server 7 with SP 4 Installed
- Windows 2000 Server with SP 4 installed
- Microsoft Access 2000
- Intel Dialogic System Software Version 5.1.1
- Parity VOS Version 8.2 (runs the telephony app)
- Symantec pcAnywhere 10.5 (Host only)
- WinZip 8.1
- VNC (Virtual Network Computing) Version 3.3.3
- Dell Array Manager 3.2

All components were installed per existing installation instructions. The telephony system was tested and deemed functional according to existing testing standards.

Background - Issues and Solutions

I should begin by noting that the creation of security policies is critical to the success of any security project. However, due to the timeframe and scope of this project, the security policies required will be written after the project is completed. This is not the preferred method of implementing a security project, but as this is a case study (and not a theory paper) it is a reality I must live with.

Many of the security issues within the system were blatantly obvious and have existed for many years. These issues have not been corrected because all existing telephony systems do not reside on customer networks; therefore, they were inherently secure as long as the physical location was secure. For the sake of completeness I should note that the security of the telephony application has previously been evaluated and deemed secure. The application security is beyond the scope of this paper as it is meant to address the security of the system as a whole in regards to network integration only.

The following table shows the issues that were known prior to the project initiation. A portion of my task will be to prove that the issues identified by peer and third-party review are indeed risks.

1. Blank administrator password
2. Data backup
3. Disaster recovery planning
4. Drive shares
5. pcAnywhere security
6. RAS (Remote Access Service) security
7. SQL logon account (account/password)
8. SQL server database security (anonymous connections)
9. Virus protection
10. Windows patches/updates

After completing the SANS GSEC training (Los Angeles Sept/Oct 2003) I took the existing list and added the following items based on my existing security knowledge and training.

1. File system security (NTFS access control lists)
2. No firewall
3. Default Windows local security settings
4. No IDS/IPS
5. System logon account/password (should not use administrator)
6. Windows left unlocked (when idle)
7. No system baseline
8. Peripheral & system security (modems/ reboot units)
9. Many unnecessary Windows services running
10. No scheduled maintenance (defrag/ temp deletion etc)
11. Client/server application security

As with most projects (like this paper) there is a completion deadline. I was pushed by management to have a "first-run solution" of the security issues by the end of March 2004 and a finalized version, which covered all issues, by the end of June 2004. This relatively short time frame forced me to break the project into phases. I made the decision to begin by addressing all items that could be resolved quickly and with minimal investment. The purpose for this approach was that any costly purchases would require a lot of research and justification and I knew that process would waste valuable time. Essentially I wanted the solution that would give me the most impact in the shortest time and could be pushed out to 80 remote servers within a few weeks. First I had to create and finalize a list of issues and how to remediate those issues. I then sought peer and management approval and input. Once that was done I broke down the list into the manageable phases.

The following is the list that I compiled and released for review on January 27, 2004:

#	**Issue**	**Solutions**
1	Problem: Administrator Password	Solution: >24-character non-complex passwords
2	Problem: Backups	Solution: Ensure every server has a working backup method.
3	Problem: Disaster Recovery	Solution: Unknown
4	Problem: Drive Shares	Solution: Remove all file shares (including Admin)
5	Problem: No protection of files at rest (EFS)	Solution: Apply Windows 2000 EFS to selected directories
6	Problem: File System Security (NTFS Permissions)	Solution: Apply NTFS permissions to all critical directories adhering to the rules of "least privilege"
7	Problem: Firewall	Solution: Ingress & Egress Firewall with good logging
8	Problem: Default Local Security Policy	Solution: Apply CIS Gold Standard Security Policy
9	Problem: No protection for worms/0 day attacks	Solution: IPS System (Cisco Security Agent?)
10	Problem: Logon Account/Password	Solution: All accounts should be different and specific to the function. The telephony system logon account should be a least privilege user account
11	Problem: File system change protection	Solution: Utilize tripwire or another MD5 checksum utility
12	Problem: OS is unlocked	Solution: Lock the OS
13	Problem: pcAnywhere security	Solution: Research and implement the best pcAnywhere encryption methods.
14	Problem: Performance Monitoring	Solution: Perform regular performance monitoring using the task scheduler
15	Problem: Physical Security	Solution: If possible the system door should be locked. Research security for peripherals.
16	Problem: RAS Security	Solution: The logon RAS account password should change regularly.
17	Problem: Reboot unit security	Solution: Change default security code on reboot units
18	Problem: Default System Services	Solution: Remove unnecessary system services.
19	Problem: No scheduled maintenance	Solution: Create a maintenance schedule for the SQL DB as well as service packs/patches. Research automation of system tasks (defrag/temp file deletion etc)

20	Problem: No log monitoring	Solution: All log files created by Firewall, IPS, IDS, NT Event Viewer should be reviewed. Research log file automation/consolidation.
21	Problem: SQL Logon Account	Solution: The application should have its own SQL logon account. The sa account password should be hardened. The telephony account password should change and should not be hard coded into the app.
22	Problem: SQL Server Security	Solution: Port filtering (with firewall) to SQL ports by approved IP only.
23	Problem: Virus Protection	Solution: Unknown
24	Problem: Client Server Application Security	Solution: IPSEC/VPN or SSL connection to the server from the client workstation.
25	Problem: Windows Updates/Patches	Solution: Internet connected servers should be updated monthly, non-connected servers should get a CD monthly or quarterly.

The response to these solutions was not what I had anticipated. Management as well as technical staff agreed with all the issues outlined; however, I did not receive any additional comments or ideas as expected. The lack of enthusiasm for the project left me burdened with the task of confirming my own hypothesis and answering many of the unresolved issues on my own.

With my deadline fast approaching, I broke down the tasks that I felt were critical for the phase one rollout. I wanted to ensure that all systems added to customer networks would survive any security issues and any security audits. The following is the list of items chosen for phase one of the project:

1. Administrator/Administrative account / password
2. Backups
3. Drive shares
4. EFS (Encrypted File System for files at rest)
5. File system security (NTFS access control lists)
6. Firewall (egress and ingress filtering)
7. Gold standard security settings
8. Logon account/password (not administrator account)
9. OS locked (when idle)
10. pcAnywhere security (encryption, etc.)
11. RAS security (changing dial-in passwords)
12. Remove unnecessary services
13. Scheduled maintenance (defrag/ temp deletion etc)
14. Windows patches/updates

Paring the original list of 25 items down to 14 was not easy, but I knew that IPS/IDS would require a lot of money, research, and training so they did not fit into the time frame. I knew that any changes to SQL server security would require the time of engineering staff in the form of code changes and would not be completed in time. I also left out issues that had no obvious resolution such as antivirus and disaster recovery. Initial research proved antivirus software and disaster recovery policy would be a politically charged issue because most customers already have their own systems in place. Log monitoring and

performance monitoring were also excluded for the sake of time and money. In regard to physical security it was determined that customers often had to reboot the server or swap backup tapes which required the system to remain unlocked.

Assessing the Risks

Although most of the issues outlined in the tables above have been known for years to be security issues, I have set about to prove these presumptions against our telephony system. As all systems are not created the same and every custom software solution creates its own set of requirements, I felt it prudent to test against them all.

Issue One – Blank Administrator Password
It didn't take a lot of research to find out why this was a bad idea; however, once proven it's easier to make the case for change. First, there are the worms/viruses that spread through the use of blank administrator passwords. A search of SARC (Symantec Antivirus Research Center) brought up 6 such worms that attempt remote connections using the administrator account with a blank password. A few of these worms are: Backdoor.IRC.Flood.E[1], W32.Randex.B[2], Backdoor.IRC.Aladinz.D[3]. At greater risk is the compromise of the administrator account that has complete control of the system. Without a password set on this account the system is by default compromised since nothing has to be "proven" to gain access to the system. If any user can log on as administrator without having to know the password, all additional security placed on the system becomes useless. My concern with this item was that curious customer staff could browse the network and easily gain access to the entire system under the context of administrator. I am also concerned about network aware worms/viruses connecting to the system.

Issue Two – Backup
The best phrase I have ever heard regarding backup is "It's not a matter of 'if' you will need it, but when". I'm not sure who said it first, but it's been spoken so much that most mid-sized and larger organizations all take backup very seriously. Without a good backup system, you have no way to recover from hardware/software failures. It also negates any disaster or business continuity plans you may have in place. My concern with this item was the loss of an entire system due to lightning strike or hardware RAID failure. Historically our company has seen about 1 to 2 systems have a catastrophic failure each year, and without proper backup all data would have been lost.

[1] http://securityresponse.symantec.com/avcenter/venc/data/backdoor.irc.flood.e.html
[2] http://securityresponse.symantec.com/avcenter/venc/data/w32.randex.b.html
[3] http://securityresponse.symantec.com/avcenter/venc/data/backdoor.irc.aladinz.d.html

Issue Three – Drive Shares
The system currently has default drive shares (C$, D$ etc.) and drive shares that are created at the time of install (Croot, Droot). The permissions on these shares are 'Everyone', 'Full Control'. Obviously this falls under the same category as the blank administrator password. My concern with this issue is that any LAN user browsing the network could reveal the shares, and since they are not protected, anyone (literally) can delete or create data on the system.

Issue Four – EFS (Encrypted File System)
The sensitive nature of the data on the system presents a risk to any of our servers. The fact that the customer owns the hardware and always has an option to stop using the application presents a risk in that the server can be repurposed. If the system were to be sold or stolen this could also be an issue. The use of EFS (Encrypted File System) will ensure that the data could not be read or recovered if any of these scenarios did occur. An additional benefit to using EFS is that since the files would be encrypted under the context of a specific user, any other user account would not have access to these files. This is beneficial in a scenario where the system was compromised utilizing a lower level account such as a user or backup operator.

Issue Five – ACLs (Access Control Lists)
The use of NTFS ACLs will help to increase the overall security of the system. The risk again is that default windows ACLs are set to 'Everyone' 'Full Control' which essentially means any authenticated user (even Guest!) has full access to all files on the system. My concern with this item is that with the use of network shares present any user on the network could connect to the system and make file modifications. By applying the appropriate ACLs to the system we are ensuring that the least amount of privilege is given to each account and that only the accounts that need access have access to specific files.

Issue Six – Firewall
Many of the issues we have discussed thus far have hinged upon uninhibited access to the server from the network. Our first line of defense and the best way to prevent unauthorized connections of any kind will be to implement a software firewall. The risk of not utilizing a firewall is: network aware worms/viruses, malicious or curious network users, and hackers who have gained access to the network. All of these create issues of data integrity and system stability. My concern with not having a software firewall is that we do not know the state of the networks we will be joining. Each of the 80+ networks where the servers will be connected will all have different dynamics as far as security and overall risk. The reality is that most IT organizations will probably have a firewall protecting the perimeter of the network and will not allow unauthorized traffic to travel the LAN; however, we cannot guarantee this. It seems prudent to protect the server itself with a software firewall to ensure that only authorized connections are made.

Issue Seven – Local Security Policy

The default local Windows security policy leaves a lot to be desired. I'll have to be honest in that in the past I have manually modified many of the settings within the Local Security Policy. Some of these include the default password policy and lockout policies. Since taking the GSEC training course I realized that templates were a better way to operate. My concern is obviously that settings we attempt to make as far as tighter passwords, etc. could be circumvented because the system would not "require" those settings. To prevent that from happening I plan to apply the CIS (Centers for Internet Security) Gold Standard security templates which activate a lot of great security features already contained in Windows. The Gold Standard template has been proven to increase the overall security of a Windows system. [4]

Issue Eight – Logon Account/Password

Adhering to the rule of "least privilege" or the practice of "least privilege administration"[5] helps to ensure a more secure computing environment. It is very common to create users/operator accounts on a system and leave them set to the default security setting of Administrator. The risk of doing this is that if the operator or the application were to perform any malicious activity, intentional or not, it would succeed under the context of a "super user". By utilizing the rule of least privilege, the application and logon account can be created with only the permissions necessary to perform the intended functions. The theory I must prove is: can the server run under the context of a standard user account with the least privilege possible? My biggest concern with this item is that a user will inadvertently delete critical system files or launch a virus or other malicious program on the system. By adhering to the rules of least privilege neither of the actions would be possible under the context of the standard user account.

Issue Nine – OS Locked

The reality of the server system that our company has designed is that it must remain logged in. There are many reasons why it must remain logged in but that really is not at the heart of this paper. The big question is: what is the risk and how can we help secure it? The risk with this is really accounting and access-control. If the system is left logged in and any user (within the secured environment) at the customer site can interactively manipulate the system we will not have any idea who that person is or what they have done. It seems trivial, but I feel that it is necessary to take this issue very seriously and ensure that the system "locks" after a period of time. This Windows feature is available in 2000 and XP by pressing CTL+ALT+DEL then pressing the Lock Computer button.[6] This will ensure that only a person that knows the logon account/password or an administrator logon will be able to unlock the system. Utilizing the features within pcAnywhere and having a timed screen-saver that locks the system, we can keep the system locked at all times when it is not in use.

---

[4] http://www.landfield.com/isn/mail-archive/2003/Feb/0002.html
[5] http://www.clocktowertech.com/newsletters/200311/server.htm
[6] http://www.microsoft.com/WindowsXP/expertzone/columns/honeycutt/03february03.asp

Issue Ten – pcAnywhere Security
pcAnywhere security is not so much a risk as it is a research project. Those before me and likely after me will want to ensure we are utilizing all the security options we can within this application. In the past our company has always used default settings, however, it has been known that there are additional settings available which help to strengthen the security of pcAnywhere. My only concern here is that a customer of ours will have concerns about pcAnywhere running on the network that we cannot address well. By better understanding the product and being able to explain (and implement) all the security features of the product, we can better inform the customer of how it will be used and secured.

Issue Eleven – RAS (Remote Access Service) Security
Remote access security on these systems has been very relaxed for a long time. Currently, the RAS username and password is the same on every server and does not change. Weak username/password combinations for RAS could be compromised using a password cracking technique such as Demon Dialing.[7]
My concern with this item would be that a former employee would decide to connect to these remote servers having previous knowledge of the username/password combination. From this point the offender could do some type of damage to these systems with complete obscurity. There is also the issue of a server-wide compromise should the username/password ever be leaked outside our company.

Issue Twelve – Unnecessary Services
The memory of the CodeRed[8] worm comes to mind when I think about unnecessary services. To quote an article on securing web services from Microsoft's web site:

> *Windows services are vulnerable to attackers who can exploit the service's privileges and capabilities and gain access to local and remote system resources. As a defensive measure, disable Windows services that your systems and applications do not require.* [9]

The reason the CodeRed worm was so successful was because many people were running vulnerable web servers and did not even know they had a web server at all. By default the web server service was running even though it was not being used. Obviously you would not turn on an appliance in your house like a television or a stereo if you were not going to use it. The same should be true on your computer, especially a critical application server such as ours.
My concern with this item is that a worm or virus will exploit a vulnerability in a service we do not need and the system will be compromised. The added bonus of turning off unnecessary services will be to free up system resources in the form of memory and CPU usage that are used by these services.

[7] http://en.wikipedia.org/wiki/Demon_dialing
[8] http://www.symantec.com/avcenter/venc/data/codered.worm.html
[9] http://msdn.microsoft.com/library/default.asp?url=/library/en-us/secmod/html/secmod89.asp

Issue Thirteen – Scheduled Maintenance
Scheduled maintenance is less a security risk than a performance risk. Obviously if your system fails due to poor maintenance, then all the security in the world won't help you. The issues to be addressed are: drive defragmentation, temp file deletion, and possible scheduled reboots of the system. All of these items will ensure the continued successful operation of the system.

Issue Fourteen – Windows Patches/Updates
Ensuring that Windows patches and updates are current help to keep the system functioning properly and secure. The biggest issue will be determining who is responsible for keeping the systems current. Many IT organizations force all vendors to allow them to update vendor systems, yet others will require our company to do it. The risk is that, for example, the system does not have the latest security patches installed and a worm/virus is released that utilizes a known vulnerability that infects the network where the system resides. The system would be open to attack due to the fact that it had not been patched. Recent examples of this would be the Blaster[10] and Sasser[11] worms that have flooded the Internet in recent months. Keeping patches up to date will be another layer in our "defense in depth"[12] strategy.

Implementation

Having narrowed my list to fourteen critical items and proven through my research that these items were indeed risks, I was ready to work on getting the Hackme system secured.

Step One – Administrator Password
Step one was to change the administrator password. I decided to be creative and I did random keystrokes on the keyboard to create a super complex password complete with numbers and symbols. I made careful note (in a secure location) of this 40+ character complex password then proceeded to log out of the system. That was the last time I was able to log on to the system. I never was able to replicate my password. I'm not sure if it was spacing issues or not being able to recognize the myriad of symbols in the password. So step one led to step zero, I formatted the system, built it back up and started back at square one.

Step one (revisited), this time I was much more methodical about my approach. I left the existing administrator account intact and created a new administrator account on the Hackme system and created a pass-phrase utilizing lowercase letters only. In fact the new password was a 40-character sentence that was

[10] http://securityresponse.symantec.com/avcenter/venc/data/w32.blaster.worm.html
[11] http://www.sarc.com/avcenter/venc/data/w32.sasser.b.worm.html
[12] http://en.wikipedia.org/wiki/Defense_in_depth

both easy to remember and easy to type. After doing a quick Google search on the topic of brute force cracking, I discovered an article where the author had calculated that with a Pentium class machine it would take approximately 15,000 years of continuous operation to crack all of the password combinations in an eight-character password. An eight-character password using all keyboard combinations of characters (84) creates two quadrillion possibilities[13]. Based on this I feel very confident that the 40-character passwords I've chosen will never be compromised. The following accounts were given long pass-phrases:

1. Administrator (system)
2. Administrator account (created)
3. User Account (created)

The only account that was not changed initially was the RAS account since it was determined that it would require a change in many of the automated systems that dialed into the remote servers. However, as the rollout is completed, all the RAS passwords will be changed at the same time.

Step Two - Backup

This really became a policy issue. Essentially the customer owns the data created on the system so therefore we cannot extend our corporate backup policy into the customer realm. However, having a good backup of existing data is critical to the value of the system. It was determined that if any facility was not doing regular backups of the data through the installed tape drive, we would keep a tape in the drive and backup as much as possible. The backup will consist of the database and as many of the current voice recordings as will fit on the tape.

Step Three – Drive Shares

There is a client/server application that allows users of the system to search and listen to previously recorded calls. This system currently uses unrestricted drive shares for file access. This connection method will be addressed in more depth with the phase two rollout. It will likely utilize an SSL or IPSEC connection to connect clients to the server. The application rollout will not commence prior to the completion of phase two, so it was decided to remove all drive shares from the system immediately. I removed the following shares from the Hackme system:

1. IPC$ (System share)
2. ADMIN$ (System share)
3. C$ (System share)
4. D$ (System share)
5. DriveC (Created share)
6. DriveD (Created share)

[13] http://www.cs.umn.edu/help/security/brute-force-cracking.html

Step Four – Encrypted File System

The next step was the decision to implement EFS on the voice recordings to keep them safe from system repurposing or in the event that there is a system compromise. I also chose to encrypt the directory where the SQL database is backed up. I encrypted the files under the user account that I had created earlier using Windows Explorer. From this point forward the process became very challenging. I logged back into the system using the user account that I had created and attempted to start the application. It failed to start due to a permission problem, which led me down the winding path of the RunAs utility. I found the Windows 2000 RunAs utility to be very difficult to use and found that it lacked the features I needed. I then downloaded and implemented the RunAs Pro utility created by MAST Software, which allowed me to easily run my application as the administrator[14]. The RunAs Pro utility also allowed me to save my RunAs parameters (including password) in an AES encrypted file that I could easily launch. I was off to the races and the testing began; this is where things got ugly. The operation of the system had problems right away and it suddenly dawned on me that since I was running the application under the administrator account it no longer had access to the files I had encrypted under the user account. I unencrypted the files under the user account, re-encrypted them under the administrator account, and tried again. Finally the system functioned and I was able to put in a few tests and listen to them for problems. Yes, there were more problems. To date I don't know why but there were certain voice files that had previously existed on the system which played without incident. However, if I attempted to record new voice files in that portion of the application I received several errors regarding permissions. My assumption is that there is a portion of the application that calls another process utilizing different permissions that I cannot control without code review. I lost about one full day trying unsuccessfully to get EFS to work; therefore, the decision was made to abandon EFS for now. Performance on a fully loaded system running EFS with database and telephony functions may also be an issue. However, since EFS did not work, there was no need for testing.

Step Five – NTFS Permissions

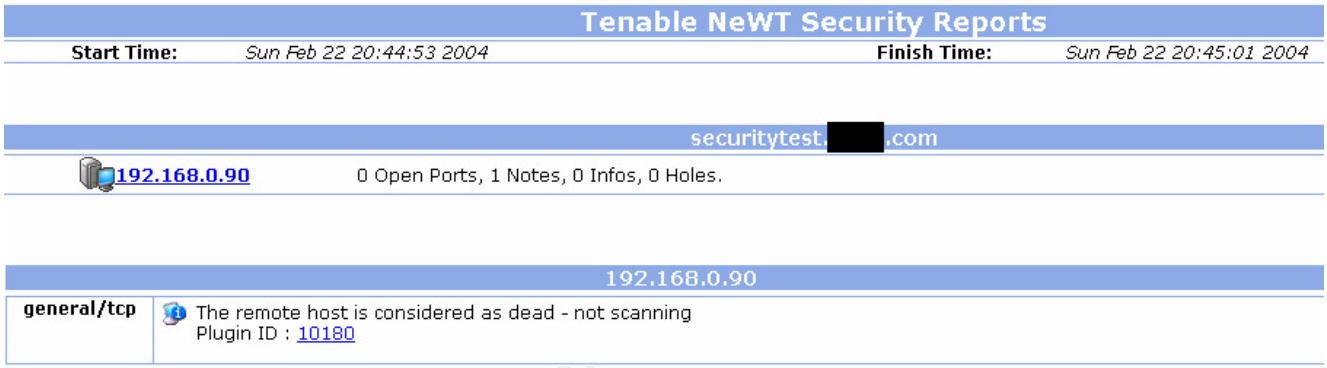
I had to ensure that the proper NTFS file permissions were applied to all directories that had been deemed critical. I removed the "Everyone" group from the voice recordings directory and the database backup directory. I then added full permissions for the administrators group, administrator account, user account, and RAS account. Testing of the system proved that these changes had no adverse effect on the application.

Step Six - Firewall

It's funny how the simplest and least complex task can turn out to be the most difficult. The word "research" roughly translates to time, lots of time. I spent about a week trying different products, talking to vendors, downloading demos, testing, uninstalling, and retesting. The madness finally landed me at the ISS

[14] http://www.mast-computer.com/c_9-l_en.html

(Internet Security Systems) BlackIce product[15]. BlackIce Personal Firewall was cheap, simple to install, and effective. The downside is that it does not have all the reporting capability I wanted or built-in egress filtering. After much research, I have resigned myself to the fact that the exact firewall I want does not exist. A software firewall is required for this project because a hardware firewall was not an option. These upgrades/updates will take place remotely and existing systems are currently rack mounted in customer datacenters. With the firewall installed, I merely had to adjust the rule set to allow the traffic from the RAS connection and I was off and running. The final test of the firewall was to hit it with a port scan. I chose NeWT (NESSUS W32 port) and chose to do a full scan of the host using all (except dangerous) plug-ins[16]. Here are my results.

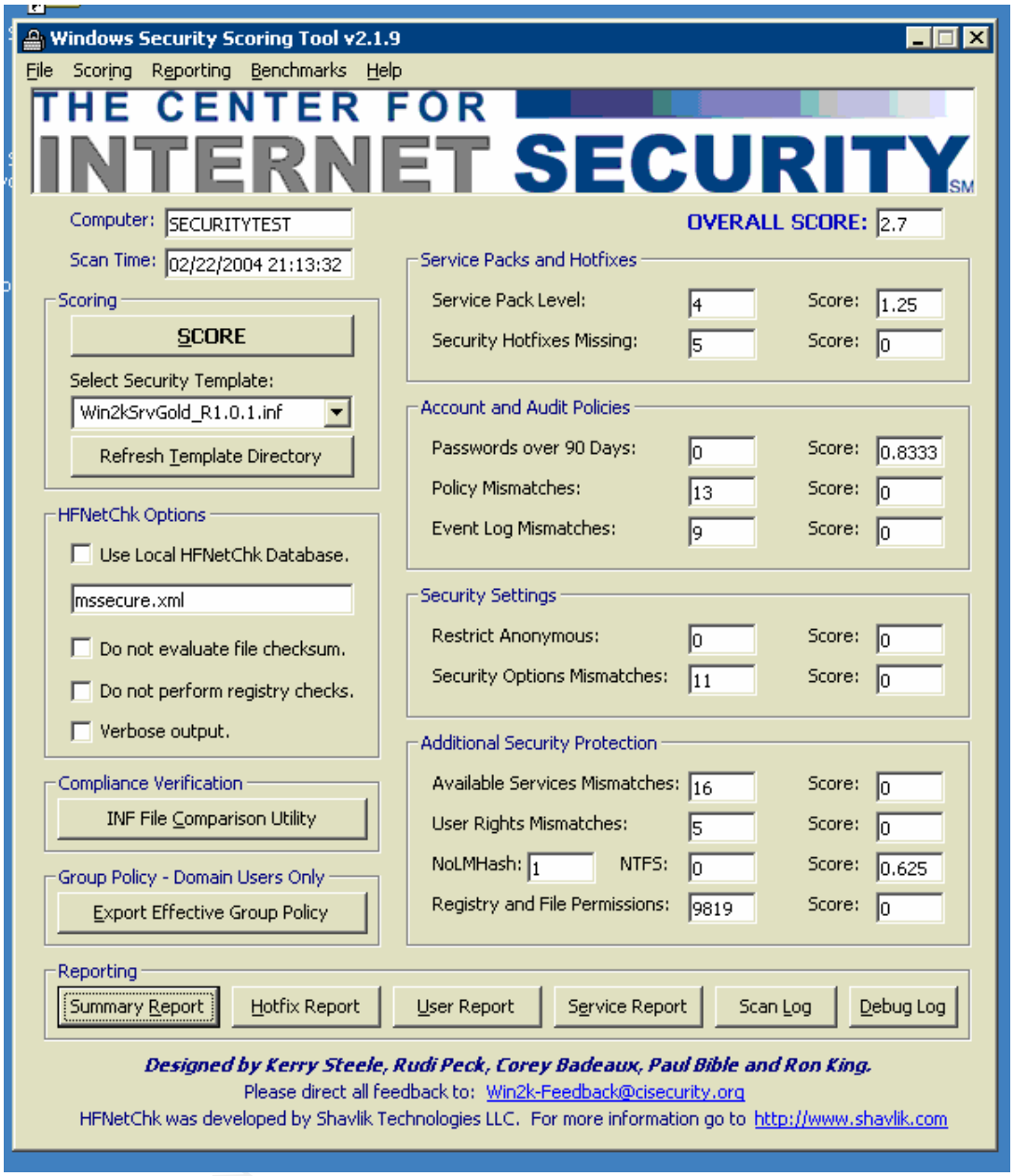


As expected the host does not show up on the network at all. The firewall is blocking all incoming traffic, as it should.

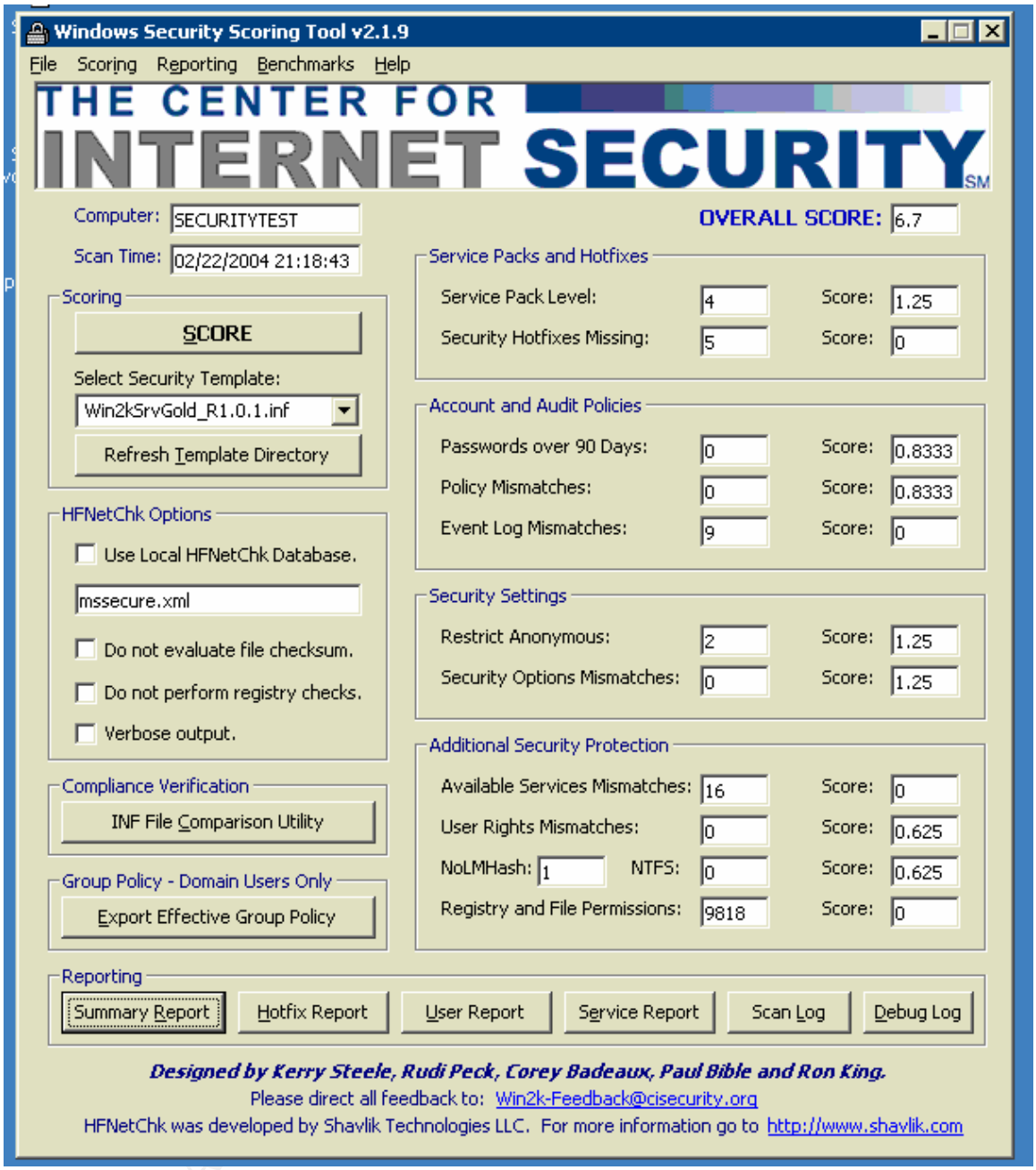
Step Seven - Gold Standard

I was very curious what difference the Gold Standard would make on the Hackme system, so I ran the CIS Windows Security Scoring Tool prior to applying the Windows 2000 Server Gold Standard and received the following results[17]:

[15] http://blackice.iss.net/product_pc_protection.php
[16] http://www.tenablesecurity.com
[17] http://www.cisecurity.org/bench_win2000.html



A score of 2.7 prior to applying the Gold Standard template left a lot of room for improvement. After applying the template I received the following:



A 6.7 is not monumental but it's still a dramatic improvement and a step in the right direction. I should note one caveat that was discovered upon testing. The Gold Standard automatically removes all accounts from the "Logon Locally" policy object leaving only the built in administrator account with this privilege. Under my proposed framework of least privilege this obviously did not work so I added the RAS account, the created administrator account and the user account.

Step Eight – Logon Account

For normal operations, the system will be logged on under the standard user account and all utilities (including the application) will be launched using a RunAs

Pro encrypted file. EFS appears to be the only insurmountable challenge created by running the server under a standard Windows user account. All other issues seem to be resolved by using the RunAs Pro utility and when necessary creating the encrypted file to launch those various utilities. I also renamed my administrator account to something more discreet. And no, I'm not going to tell you what it is!

Step Nine – OS Locked

During the investigation of step nine I had a gem of a discovery while researching the security options provided within pcAnywhere. Our support staff connect to the remote servers using pcAnywhere and typically no customers ever have a need to administer the system. Symantec was security conscious enough to include a feature that allows me to have the system lock on any pcAnywhere disconnection. I also set the default screensaver to lock after 60 minutes in the event that customer personnel logged into the system (typically for backup) and did not lock it when finished. I feel this provides adequate system protection when idle.

Step Ten – pcAnywhere Security

Step nine segues right into step ten, which is pcAnywhere security. I think there is a terrible misperception about the product in general. It is often thought that pcAnywhere is a hacking tool that is often misused when, in reality, it is actually an invaluable tool for troubleshooting remote systems. I looked at some of the default settings that may better protect our connections. I discovered the relatively light but "better than nothing" setting for encryption of the pcAnywhere connection. Symantec describes the encryptions settings as:

> **pcAnywhere encryption:**
> Of the three levels of encryption supported in pcAnywhere, pcAnywhere encryption is the least secure. pcAnywhere encryption scrambles the data stream, using a simple mathematical transformation, so that a third party cannot easily interpret it. The data is sent in non-clear-text format. It is designed to prevent someone from reading the pcAnywhere data stream and immediately knowing what is being transmitted. However, if the data stream is captured, a cryptographer could break the encryption without too much effort.
> pcAnywhere encryption is intended for users who do not have access to a cryptographic service provider or who want to connect to a computer that uses an older version of pcAnywhere that does not support a higher level of encryption. [18]

I chose to implement the pcAnywhere encryption setting because it was simple (point and click) and did not require a public key infrastructure. The other system locking measures were already covered in step nine. Although this encryption setting is extremely light, we currently connect to the systems through RAS (Remote Access Service) directly into the system. In the future we will likely be connecting to servers through site-to-site VPN tunnels (IPSEC) so this encryption

[18] http://service1.symantec.com/SUPPORT/pca.nsf/pfdocs/2001060508510012?Open

setting meets my need for protecting the data once it leaves the VPN tunnel and crosses the customer LAN.

Step Eleven – RAS Security
As previously discussed the RAS password change will not be implemented until the final rollout to the last server is complete. This will likely occur after this paper is finished; however, I already know I will create a pass-phrase of at least 40 characters. The use of the RAS account is only performed by automated systems to make connections to the server so I will likely choose a more complex password utilizing letters, numbers. and symbols.

Step Twelve – Unnecessary Services
Step twelve proved to be enlightening and a very exciting challenge. My Google research led me to a couple of websites which helped me compile a list of services that I did not need to have running. I looked at every service on the system that was set to start up automatically then compared it to several lists I found at blackviper.com[19], Microsoft.com[20] and techspot.com[21]. The following is a list of services that I decided to disable based on my research:

- Alerter Service
- Server Service
- Remote Registry Service
- License Logging Service
- Distributed Link Tracking Client
- Computer Browser
- Messenger
- SMTP
- SNMP
- TCP/IP Netbios Helper Service
- WWW Publishing Service
- Distributed File System

These services were set to manual:

- Print Spooler
- Automatic Updates
- System Event Notification
- Windows Management Instrumentation
- Windows Management Instrumentation Driver Extensions
- The telephony application (RunAs utility issue)

The telephony application could no longer be run as a service because of the use of the RunAs utility. Attempts to start the service under the new administrator account failed so it was decided to proceed without the application running as a service. I also downloaded the sc.exe utility that is part of the Windows 2000 Resource Kit[22]. The sc.exe utility allowed me to create a batch file to change the

[19] http://www.blackviper.com/WIN2K/servicecfg.htm
[20] http://www.microsoft.com/technet/prodtechnol/windowsserver2003/proddocs/deployguide/iisdg_sec_rmkz.asp
[21] http://www.techspot.com/tweaks/win2k_services/print.shtml
[22] http://msdn.microsoft.com/library/en-us/tools/tools/service_control_utility.asp

startup mode for all of these services. This will save me time as I roll out phase one to all remote servers.

Step Thirteen – Scheduled Maintenance
Step thirteen began to get mired in the application of version releases and other "scheduled" updates and has been lost in a black hole discussion about customer notification. Alas I'm forced to remove it from the phase one rollout and push it into phase two. Hopefully a consensus can be reached soon.

Step Fourteen – Windows Updates/Patches
Step fourteen has been decided simply on bandwidth requirements for Windows updates and service packs. The following is an overview of the policy I created:

- All networked systems with Internet access will be updated monthly.
- All networked systems without Internet access will be updated quarterly.
- All non-networked systems will be updated yearly with service packs only.

I feel this approach will keep the most vulnerable systems current on hotfixes and the least vulnerable systems will at least be getting the OS updates they need. This method also prevents creating too much burden on existing technical staff.

Conclusions

So, was all the work worth it? To date we have five systems that have been added to customer networks where the phase one rollout has been applied. Three of these locations have also been connected via site-to-site VPN instead of the standard dial-up connection since this is more secure for both parties involved. At one location the security administrator performed a vulnerability assessment of the system which I will include below. This first scan was before the security measures were applied and was performed using Nessus with all available add-ins[23]. For the sake of the reader I'll only include here the open ports and high risk items:

Network Vulnerability Assessment Report **26.02.2004**

Sorted by host names

Session name: Telephony Server	**Start Time:** 19.02.2004 13:32:16
	Finish Time: 19.02.2004 14:00:28
	Elapsed: 0 day(s) 00:28:11
Total records generated: 72	
high severity: 1	
low severity: 39	

[23] Anonymous Customer Location – Nessus Scan

informational: 32

Summary of scanned hosts

Host	Holes	Warnings	Open ports	State
10.150.1.80	1	39	32	Finished

10.150.1.80

Service	Severity	Description
unknown (1048/tcp)	Info	Port is open
loc-srv (135/tcp)	Info	Port is open
netbios-ssn (139/tcp)	Info	Port is open
microsoft-ds (445/tcp)	Info	Port is open
java-or-OTGfileshare (1050/tcp)	Info	Port is open
unknown (1051/tcp)	Info	Port is open
unknown (1055/tcp)	Info	Port is open
unknown (1057/tcp)	Info	Port is open
ms-sql-s (1433/tcp)	Info	Port is open
pptp (1723/tcp)	Info	Port is open
pcAnywheredata (5631/tcp)	Info	Port is open
loc-srv (135/udp)	Info	Port is open
netbios-ns (137/udp)	Info	Port is open
netbios-dgm (138/udp)	Info	Port is open
microsoft-ds (445/udp)	Info	Port is open
isakmp (500/udp)	Info	Port is open
unknown (1027/udp)	Info	Port is open
unknown (1044/udp)	Info	Port is open

ftp (21/tcp)	**Info**	Port is open
unknown (1049/tcp)	**Info**	Port is open
unknown (1050/udp)	**Info**	Port is open
unknown (1052/tcp)	**Info**	Port is open
unknown (1060/tcp)	**Info**	Port is open
unknown (1132/udp)	**Info**	Port is open
datametrics (1645/udp)	**Info**	Port is open
sa-msg-port (1646/udp)	**Info**	Port is open
l2tp (1701/udp)	**Info**	Port is open
radius (1812/udp)	**Info**	Port is open
radius-acct (1813/udp)	**Info**	Port is open
unknown (2148/udp)	**Info**	Port is open
IISrpc-or-vat (3456/udp)	**Info**	Port is open
pcAnywherestat (5632/udp)	**Info**	Port is open
ftp (21/tcp)	**High**	It may be possible to make the remote FTP server crash by sending the command 'STAT *?AAA...AAA. An attacker may use this flaw to prevent your site from distributing files *** Warning : we could not verify this vulnerability. *** Nessus solely relied on the banner of this server Solution : Apply the relevant hotfix from Microsoft See:http://www.microsoft.com/technet/security/bulletin/ms02-018.asp Risk factor : High CVE : CVE-2002-0073, CVE-2002-0073 BID : 4482

There are obviously quite a few open ports and it's obvious that the system needs some additional security measures in place. The following is the scan performed by the customer after the phase one measures were in place.

Network Vulnerability Assessment Report **26.02.2004**

Sorted by host names

Session name: TelephonyServer	**Start Time:** 19.02.2004 15:35:06 **Finish Time:** 19.02.2004 15:58:37 **Elapsed:** 0 day(s) 00:23:30

Total records generated:	1
high severity:	0
low severity:	1
informational:	0

Summary of scanned hosts

Host	Holes	Warnings	Open ports	State
10.150.1.80	0	1	0	Finished

10.150.1.80

Service	Severity	Description
general/udp	Low	For your information, here is the traceroute to 10.150.1.80 : 10.150.10.6 10.150.10.3 ?

As you can see the host was essentially “invisible” primarily because of the software firewall blocking all ports. If the firewall were to fail we now have appropriate measures in place to protect the system. With NTFS permissions and obeying the rules of “least privilege”, the system will remain secure.

As phase one comes to a close I look forward to implementing the remainder of the security solutions identified for phase two. I know that as security awareness grows, I will look to the security community for additional training, information and support.

REFERENCES

1. Neal Hindocha “Backdoor.IRC.Flood.E” November 16, 2003. URL:http://securityresponse.symantec.com/avcenter/venc/data/backdoor.irc.flood.e.html (05 May 2004).
2. Douglas Knowles “W32.Randex.B” August 11, 2003. URL:http://securityresponse.symantec.com/avcenter/venc/data/w32.randex.b.html (05 May 2004).
3. Ying Lin “Backdoor.IRC.Aladinz.D” September 19, 2003. URL:http://securityresponse.symantec.com/avcenter/venc/data/backdoor.irc.aladinz.d.html (05 May 2004).
4. Dan Verton “Free benchmark could have found Slammer vulnerability” January 31, 2003. URL:http://www.landfield.com/isn/mail-archive/2003/Feb/0002.html (05 May 2004).
5. Clocktower Technology Services, Inc. “Least Priveledge Administration” November 2003. URL:http://www.clocktowertech.com/newsletters/200311/server.htm (05 May 2004).
6. Jerry Honeycutt “Windows XP Security for Everyone” February 3, 2003. URL:http://www.microsoft.com/WindowsXP/expertzone/columns/honeycutt/03february03.asp (05 May 2004).
7. Wikipedia “Demon Dialing” May 4, 2003. URL:http://en.wikipedia.org/wiki/Demon_dialing (05 May 2004).
8. Eric Chien “CodeRed Worm” July 29, 2003. URL:http://www.symantec.com/avcenter/venc/data/codered.worm.html (05 May 2004).
9. Microsoft Corporation “Securing Your Web Server” January 2004. URL:http://msdn.microsoft.com/library/default.asp?url=/library/en-us/secmod/html/secmod89.asp (05 May 2004).
10. Douglas Knowles, Frederic Perriot and Peter Szor “W32.Blaster.Worm” February 26, 2004. URL:http://securityresponse.symantec.com/avcenter/venc/data/w32.blaster.worm.html (05 May 2004).
11. Heather Shannon “W32.Sasser.B.Worm“ May 5, 2004. http://www.sarc.com/avcenter/venc/data/w32.sasser.b.worm.html (05 May 2004).
12. Wikipedia “Defense in depth” December 18, 2003. URL:http://en.wikipedia.org/wiki/Defense_in_depth (05 May 2004).
13. *University of Minnesota* "Why Brute-Force Password Cracking is Impractical" 1997. URL: http://www.cs.umn.edu/help/security/brute-force-cracking.html (22Feb. 2004).
14. Mast Software -http://www.mast-computer.com/c_9-l_en.html

15. BlackIce Personal Firewall - http://blackice.iss.net/product_pc_protection.php

16. *Microsoft TechNet* "Enabling Only Essential Windows Server 2003 Components and Services" 2004. (22 Feb 2004)

17. Tenable Security - http://www.tenablesecurity.com

18. CIS Scoring Tool - http://www.cisecurity.org/bench_win2000.html

19. *Black Viper* "Windows 2000 Professional and Server Services Configuration 411" January 23, 2000
URL: http://www.blackviper.com/WIN2K/servicecfg.htm
(22 Feb. 2004).

20. Microsoft TechNet "Enabling Only Essential Windows Server 2003 Components and Services" 2004
URL:http://www.microsoft.com/technet/prodtechnol/windowsserver2003/proddocs/deployguide/iisdg_sec_rmkz.asp
(22 Feb. 2004).

21. *Thomas McGuire* "Windows 2000 Services Tweak guide**"** March 23, 2000**.**
URL:http://www.techspot.com/tweaks/win2k_services/print.shtml
(22 Feb. 2004).

22. Microsoft Service Control Utility - http://msdn.microsoft.com/library/en-us/tools/tools/service_control_utility.asp

23. Anonymous Customer Location – Nessus Scan. (26 Feb 2004)

Last Updated: July 20th, 2017

Upcoming Training

SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Trenton SEC401	Trenton, NJ	Aug 21, 2017 - Aug 26, 2017	Community SANS
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
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 Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS