Global Information Assurance Certification Paper

Controlling Viruses in a Fortune 500 Company
GSEC version 1.4b Option 2
James Bradley

Abstract

This case study describes the state of affairs at my company when I first came on board in the spring of 2001, the analysis of the problem relating to the prevalence of virus outbreaks, and the process employed to mitigate this constantly recurring problem. The company I work for is one line of business in a Fortune 500 holding company.

Before Snapshot

The company was being adversely impacted by numerous recurring virus outbreaks and was at significant risk of a serious outbreak that could cripple the company. The first and most obvious issue was that of lost productivity both in the business departments and in the information technology department due to the virus outbreaks that were encountered. Virus infections were also being spread to one of the other lines of business due to inter-domain trusts and access to shares on the other network. Other risks were present that will be explained below.

The business departments were losing productivity since the users could not effectively perform work while a virus was active on their computer. An additional and more significant element of risk was posed due to the nature of the company's business and the information contained in the company's computers. Our company is a federally regulated financial services organization that is subject to the Gramm Leach Bliley Act requirements to protect consumer and customer information for deliberate or accidental disclosure. We are also required to adhere to a number of other federal regulations concerning information security practices and programs. We could not demonstrate that we had an adequate program in place to mitigate risks from viruses, which is a requirement in the Federal Financial Institutions Examination Council (FFIEC) Information Security Handbook[i].

In addition to viruses, we were also seeing e-mail worms that were affecting our e-mail servers by creating additional processing load that would bring down a server with some regularity. Our e-mail gateway, which was co-located with our Internet facing firewall, would at times have such a backlog of outbound e-mail that no one could access the Internet for even legitimate purposes.

Our Information Technology department was losing productivity by virtue of having to dispatch desktop technicians to the infected machines and remediate the infection. As an additional complication, many of our desktop users are in remote locations requiring expensive contract assistance be dispatched. Network administrators providing support for our e-mail servers were also precluded from performing maintenance and new

installations when having to resurrect an e-mail server that had gone down. Another hit on productivity was that whenever files would get infected, the network support staff would have to recall offsite backup tapes to get back to an uninfected version of files that the business department users needed. In addition to the time wasted in this repetitive effort, there were tangible, hard-dollar costs associated with retrieving tapes. Having the back up tapes on-site presented an additional layer of risk in that if a disaster were to occur while the tapes were not safely stored offsite, we would have severely limited our ability to recover data files that the business units would have needed. While the backup procedures, including the tape rotation schedule, were adequate to provide backups due to file corruption, damage, or loss, there are inherent limitations on the ability of the system to produce up to the minute currency of the restored files. Therefore, there were many instances where numerous man-hours of productivity from the business unit were lost to the virus.

The anti-virus product that we used was a well respected, but not well known in the U.S., commercial product – Sophos Anti-Virus (http://www.sophos.com/).

The Sophos anti-virus product is aimed solely at commercial enterprises. It does not offer any home user products. The Sophos virus detection engine consistently performs above others in performance testing measuring false positive, false negative, speed etc and has a low performance impact. Other companies such as Amerada Hess, The Bank of England, GlaxoSmithKline, Harvard University, Siemens and other large companies around the world use it.[ii]

It had a significant shortcoming in that there was no centrally managed update process to deploy new signature files or updated anti-virus engines. Nor did it have a method of reporting back on viruses that were found in the environment.

The manual burden on our network administrators to deploy these updates to over 400 servers and to 3000 plus desktop locations was prohibitively onerous. The norm at that time was to never be completely current on either signature files or engine versions. Updates were deployed using a convoluted series of lengthy batch files that routinely failed to perform as expected and provided no feedback on what was actually accomplished. The failure of the script to complete was routinely overlooked and the only way to check the status of the updates was to manually look at the files on each server.

The "Melissa" and "I Love You" viruses were significant events on our network. Even though they were old (Melissa – March, 1999[iii] and I Love You - May, 2000[iv],) with virus identification files deployed to the best of our ability at that time, they kept on recurring in our network proving the inadequacy of our anti-virus system.

Melissa, or W97M_Melissa is a macro virus that is propagated as an e-mail attachment. The e-mail would be addressed from an infected user with a title of "Important Message From (name)". Microsoft Word 97 and 2000 would allow the virus to spread if macros were enabled in those software products. The first action it takes is to lower the macro

security setting so it can run without alerting the recipient. It then checks to see if a particular registry key exists and if it does, that a specific value is in the key. If the key is absent or the value is not the expected value, the virus propagates itself by sending an infected e-mail to the first fifty people in the user's address book. The registry key is then created and the value populated so that it would propagate only once during a session. Then the macro infects the MS Word NORMAL.DOT default Word document template. Every new document created based on the default template is then infected also.[v]

I Love You is a malicious VB Script program. It comes in an e-mail message with the subject "ILOVEYOU", an attachment named "LOVE-LETTER-FOR-YOU.TXT.VBS and a message text of "Kindly check the attached LOVELETTER coming from me".

When the scrip executes it will:

> Create a file "script.ini" if a directory with mIRC is available and attempt to send the script to anyone active on the mIRC channel that the now infected user is on.
>
> Search for network shares with particular files (.vbs, .vbe, .js, .jse, .css, .wsh, .sct, .hta, .jpg, .jpeg, .mp3, .mp4) , and replace them with a copy of the worm. Anyone double clicking the infected file will start the worm again. [vi]

As we started investigating this problem, it was obvious that there was no means available to identify what virus activity existed on our network until they surfaced by propagation or damaging systems and needed manual removal. The reason for this was that Sophos did not have anything like an enterprise management solution that would report on the status of the deployment of the product or virus activities that it was detecting.

Approximately half of our servers were identified as CID (Central Installation Directory) servers. The other half was using the CID servers as the update source for their installations of Sophos. All workstations would receive their configuration files, IDE (Identity) files, and any updates to the anti-virus engine from an associated CID server. The process was that a workstation would go to its CID server to get the updates and to also push to the CID server the contents of a log file (SWEEP.LOG) that would be included in the servers log file. The SWEEP.LOG file contained information about what viruses were found on the workstation, the Sophos update activities or attempts to update – actually a continuous log of all Sophos activity. The non-CID servers would similarly push their SWEEP.LOG files to the CID servers.

As a result, we had a somewhat consolidated set of log files on each CID server that did contain the information about virus activity on all our workstations and servers that were served by that particular CID server. However, given that the log files were also full of data (versus useful information) about failed attempts to scan files, scan attempts, and a listing of current IDE files, it was difficult to gather the information about virus activity by manually reading the log. In addition, there were approximately 200 of these logs so that gleaning any useful information from them was impossible, using the manual methods in place at that time.

The only way to adequately determine the status of the detection engine (Sophos version) or the numerous IDE files that were present at the time was to manually check the GUI based Sophos administration program on that individual server for the configuration and version information, and to check each of three directories on the server that stored the current IDE files.

To recap, we had large volumes of data with no effective means available to harvest any useful information about what was going on regarding Sophos, the software version, IDE file updates, the actual viruses that were found by Sophos, and whether or not Sophos was successful in dealing with the virus. Trying to wrest information from each of those 200 or so log files on a daily basis was virtually impossible given the levels of manpower that it would require. The purpose of bringing out this information is not to castigate the Sophos Anti-Virus product. At the time that the product was purchased and installed on our network, none of the other players in that anti-virus market had developed large-scale enterprise grade solutions either. All users in large enterprises were basically blind to what was going on in their environments where virus activity was concerned.

Several attempts were made to consolidate all of the log files into a more manageable format, but those attempts failed at every turn as the volume was simply too great. Even if these attempts had been successful, the total volume of data that would have to be reviewed was completely unmanageable. No timely reporting could have been obtained as to what was happening on the servers and workstations from a global view, as the man-hours required would have been cost prohibitive.

During Snapshot:

In order to address some of the failings of the Sophos product, a companion product called ScanMail [vii] was purchased and a project was initiated to get it installed. ScanMail is a separate email filtering system that has the ability to scan for viruses and to block certain email attachments to prevent them from inducing viruses into the network from that avenue.

As a part of the ScanMail installation project, we identified the need to start blocking certain types of files from entering our network through the e-mail system. It was apparent to the Information Security experts in our organization, that at some time in the not too distant future a major virus outbreak would occur if we did not take proactive action to prevent it. Due to political considerations in play at the time, many of the project team members flatly refused to endorse such a step due to the inconvenience that it would create for the business unit users of the network and e-mail system. As a result, there were some extensions blocked, but the most important extension of .exe was not blocked.

Shortly after that the ScanMail filtering system was place online and operational, the Nimda[viii] virus was released on the Internet. Since one of the propagation methods Nimda used was to email itself in the form of an executable (.exe) attachment, the virus

bypassed our ScanMail system and landed squarely into the email box of one of our employees. This occurred approximately 6 hours before our anti virus vendor, Sophos, released an IDE file that would identify the virus. The employee opened the executable attachment initiating an extremely fast moving virus infection. In very short order, approximately 75% of our workstations were affected by it. Our web servers did not take any hits as we had recently patched them against Code Red, which exploited the same vulnerability as Nimda.

Nimda was a worm technically speaking. It is a mass-mailing worm that uses multiple methods to propagate itself. It infects local files and network shares in that it is a network aware worm in that it searches for available shares accessible from the infected machine. On web sites, it takes advantage of vulnerabilities in unpatched IIS servers (Unicode Web Folder Traversal). It also looks for sites previously compromised by Code Red and takes advantage of the "backdoor" left by that program.

If the worm arrives by e-mail, it uses a MIME vulnerability that lets the program execute even when the recipient reads or even just previews the infected message. It was also distributed from infected web sites. The infected site would prompt the unsuspecting user to download an e-mail message that contained the worm as an attachment.

And, it also created new open shares on the infected computer plus it created a Guest user account with Administrator privileges.

In order to clean up after an infection by Nimda, Microsoft Exchange had to be patched with bulleting MS01-020 and Microsoft Internet Explorer had to have Service Pack 2 installed (for both IE 5.01 and IE 5.5) Web servers (Microsoft Internet Information Server) had to be cleaned up with a tool from any of the major anti-virus providers to eliminate the Code Red back door and then apply any of a number of patches to counter the Unicode vulnerability. (MS00-057, MS00-78, MS00-086, MS01-026, MS01-044, Win2000 SP2, WinNT4.0 Security Roll-Up Package, IIS Lockdown Tool, or URLScan) As mentioned earlier, we had already protected our web servers so we didn't have to deal with rolling the IIS patches out.

But merely cleaning it up with the vendor supplied tools and then patching systems to prevent re-infection wasn't the end of the rainbow. The possibility of the attacker having made other changes to the host machine was very high. The only way to be sure that there wasn't anything else compromised on an infected machine was to re-install the operating system and restore files from backup.

Fortunately, as shown above, Nimda did not have a malicious payload – nothing that wiped out data or flashed the BIOS ala Chernobyl. Now, as much as we in the Information Security Department would usually not say something like "I told you so", I believe it did slip out once or twice in this instance. The consequences of Nimda were enormous. This company has over 3000 users that are spread over 40 states. A large portion of the cleanup effort had to be contracted out to another company at an exorbitant hourly rate. For the machines that were local, cleaning up Nimda was manpower

intensive due to the sheer volume of infected systems. Overall, Nimda was a gross waste of man-hours and tens, if not hundreds, of thousands of dollars that could have avoided by merely blocking one additional file type, .exe as was recommended.

Meetings were held among the Information Security, Network Administration, Help Desk and Desktop groups within Information Services to discuss and develop a strategy that could be used to alleviate this ongoing threat by a full implementation of blocking certain file types as was proposed prior to the Nimda virus attack. The full list of files proposed to be blocked is shown in Appendix A of this paper.

Surprisingly, there was still some resistance and dissention among the group concerning the blockage of the email extensions. However, the ongoing battle to eradicate Nimda and the fear of additional attacks that could not be easily explained to the board, made the decision by the Chief Information Officer (CIO) a foregone conclusion. Once the policy was in place, the next step was to implement it, which proved to be easier than expected. We announced the plan via company-wide e-mail and our somewhat limited change control process for several days before implementing it and provided the Help Desk with a script to follow when the inevitable questions and complaints would come in to them. While there were some complaints from the business departments, effective techniques were developed that provided solutions to their issues. The political backlash was subtle, but severe. Two senior IT people eventually took the brunt of it and are no longer present in the organization. Part of that was the result of the political debate over the file blocking and part was over the fact that the Nimda virus was able to create such havoc within our networks.

The file attachment blocking policy was a coup in the overall ability to protect ourselves from another virus attack, but did not address the primary issues that we had with Sophos.

The next thing that had to be addressed was the inability to see or report on what was happening in the Sophos environment regarding updates, virus activity identified, and actions taken. Since Sophos did not offer at that time, any solution to the problem, it was addressed internally by the Information Security Architecture department.

An Access database was created using VBA (a form of Visual Basic) and several other forms of scripting to automate the anti-virus reporting process. The database would copy the SWEEP.LOG files from each CID server individually and import them into the Access database. Once incorporated as a part of the table structure, additional scripts would parse the log files line by line and extract the information about viruses on the network from both the workstations and servers. While the information gathered was limited by that available in the Sweep.log files, included were the workstation name, time and date, and virus found. The parsing would populate separate tables with the relevant information so reports could be generated. The reports were made available to anyone with a need to know that information; either directly from the database via Access forms, or through printed paper versions.

One drawback was that the Sophos logs do not identify the status of a discovered virus. On servers this was not a real problem in that the server could easily be reviewed to determine the virus status. Workstations were another issue as they were so numerous and spread across such a large area. In order to address this issue, the database report was designed to report a device if the same virus was found on the machine for more than one day. Repeat findings of the same virus on the same machine was taken to indicate that Sophos had not been able to deal with it. The daily workstation reports were automatically prepared for the Help Desk so they could create Help Desk tickets to dispatch desktop technicians to infected machines based on the longitudinal reports described above.

The next step was to handle the problems with getting current IDE files and engine updates to the CID servers and out to the devices that they served. Additional programming was written to read the registry entries for the CID servers that compared the Sophos configuration to a standard configuration that was stored as entries in the database. In order to check the status of the IDE files on the servers, especially the CID servers, one part of the program was written using the Sophos SGet program to download the current IDE files for the version of Sophos that we were running and to create a listing of the current IDE files. This list was then compared to the servers on the network. Non-CID servers were checked for the files they used. Each CID server has three directories – one for the server itself, one for other NT devices to receive updates from and a third for the Win9x platforms to receive their updates from. The comparisons were the basis for generating exception reports to show which CID servers did not have the full sets of files.

Now that the technical issues were in hand, we also recognized the need to educate our business users and our technical IT staff on what was going on out there in the wild about viruses. We also wanted to present a repetitious reminder about how to avoid contracting a virus through various methods such as email and web browsing.

Out first effort was to create a weekly e-mail based advisory for our technical IT staff to let them know what new viruses were loose, what they did and where to find additional information about them. We also took that opportunity to include information on recent Microsoft and Unix security vulnerabilities and the vendor bulletins about how to patch them. And of course, we also have short blurbs in there about following information security practices and procedures since they are the custodians of the company's information assets.

The next step was to get additional information about viruses, how to avoid contracting them and the counter measures that were in place in the company to protect us from them into major company newsletters and intranet article postings. There are two intranet site-based documents that we utilize – First Monday, which as the name implies comes out the first Monday of every month. There's also another section on the intranet site called Virus Updates that we manage on our own. The major publication that comes out quarterly is People and is a hard copy tabloid that is distributed around the company in all of its locations.

After Snapshot

Since the implementation of all of these measures, we have not seen any outbreaks of virus infections on the internal network in over a year. Yes, there were some prices to pay, yes it was and is today sometimes a little inconvenient if someone wants to send out or receive from a legitimate source one of the file types that we block at our e-mail gateway. However, it is patently obvious that these minor inconveniences are nothing compared to the lost productivity, lost data, and dollar costs that a large virus outbreak on our network would be. This in no way is meant to imply that we have not had the occasional virus pop up in our environment. They typically have entered through a user workstation that has visited a less than reputable web site or has loaded a floppy disk that had an infected file. But since we are now on top of current anti-virus signature files, most of these are stopped at the workstation level and do not have the chance to propagate.

The user education activities that we have undertaken provide positive reinforcement to the employees as to why these measures are in place and the benefit they provide. We have even received unsolicited comments about how useful and helpful the articles are. They also go a long way toward building our information security awareness program, which is another regulatory requirement documented in the FFIEC Information Security handbook as well as a cornerstone of any decent information security program. Information security awareness training is also addressed in the Security Management Domain in the CISSP study materials. "A strong security architecture will be less effective if there is no process in place to make certain that the employees are aware of their rights and responsibilities. All too often, security professionals implement the "perfect" security program, and then forget to factor the customer into the formula. In order for the product to be as successful as possible, the information security professional must find a way to sell this product to the customers. An effective security awareness program could be the most cost-effective action management can take to protect its critical information assets."[ix]

Our awareness program enabled us to identify an unknown variant of an existing virus. One of our remote location technical support people noticed some odd behavior on a couple of local systems. According to the messages that we keep putting out to contact Information Security if there is anything suspicious looking, he called us and reported the activity. Our intrusion detection analyst used or Intrusion Detection System to examine the traffic. We ultimately sent the captured packets to Sophos for examination and were subsequently informed that it was a new variant of an already identified virus.

I think it reasonable to say that our efforts so far have been successful in mitigating the problems that were present in the environment at the outset. I think it also reasonable to say that this is not the end but just the maturing of a process that has to be kept alive and invigorated constantly to be successful as we go forward. The other thing that I think bears mentioning is that these problems were not insurmountable. The real problem was that the "old hands" were too close to the issue for too long and could not take the opportunity to back away from it and take a fresh look. The "new hands" had the luxury

of not being stuck up against the problem and could see it from a fresh perspective. The scripting and database development skills were the only really new asset that was introduced.

There were political hurdles and costs that are unfortunate. But the teamwork that was finally organized brought this effort to its current state. Diligence and perseverance were the key success factors in this case.

Appendix A

File Types Blocked at the Gateway

HTML	ASP	ADE	ADP
BAS	CHM	CPL	CRT
DLL	HTP	HTA	INF
ISN	ISP	JSE	MDB
MDE	MSC	MSI	MSP
MST	PCD	SCT	SHS
URL	VXD	WSC	WSF
WSH	BAT	CMD	COM
EMD	EXE	JS	LNK
NWS	PIF	REG	SCR
VBS	VBE	HTM	

List of References

[i] Federal Financial Institutions Examination Council web site - http://www.ffiec.gov/ffiecinfobase/html_pages/infosec_book_frame.htm

[ii] Sophos website - http://www.sophos.com/companyinfo/customers

[iii] Sendmail Inc. website information on the Melissa Virus - http://www.sendmail.com/alert/melissa/

[iv] CERT website information on the I Love You Virus - http://www.cert.org/advisories/CA-2000-04.html

[v] Op. Cit. iii

[vi] Op. Cit. iv

[vii] Trendmicro website information on ScanMail product - http://www.trendmicro.com/en/products/email/smex/evaluate/overview.htm

[viii] Sophos web site information on Nimda virus - http://www.sophos.com/virusinfo/analyses/w32nimdaa.html

[ix] Tipton, Harold & Krause, Micki Information Securty Management Handbook 4th Edition Boca Raton, London, New York, Washington DC Auerbach 1999 pg 197

Last Updated: May 27th, 2017

Upcoming Training

SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event