Global Information Assurance Certification Paper

Slamming the door on the Slammer worm
Matthew C. Boykin
GSEC Certification Practical
Version 2.5b
July 14, 2003

Abstract

The following practical will discuss the spread of the Slammer (Sapphire) Worm and how my organization successfully avoided the Slammer Worm as well as other worms using a multi-layered security model, standard best security practices, business continuity and disaster recovery plans. Having a multi-layered security approach and or model is like having multiple layers of clothing on during the winter months. Without having those multiple layers of clothes to protect you, the cold, rain, ice, snow and wind would penetrate your thin layer of protection and create havoc on your body, making you sick and unable to work or perhaps worse. Without multiple layers of security to protect your network and organization, Worms, Viruses, Trojans and Hackers would be able to wreak havoc on your network or even worse have a long-term affect on your organizations health and welfare. Over the next few pages I will outline a proven multi-layered security model, standard best security practices, business continuity and disaster recovery plans that have provided a safe and secure IT business model that does provide confidentiality, integrity and availability to my organization.

Introduction

On January 25, 2003 the Microsoft SQL Slammer worm was detected in the wild. This was one of the latest in a series of worms that has tested security preparedness of the Internet community. The Slammer worm exploited a buffer overflow in Microsoft SQL Servers on the Internet. Specifically, the computers affected were running Microsoft's SQL Server 2000 and MSDE 2000. Twenty-four months following the Code Red worm, many organizations with Internet Systems demonstrated that they were not prepared to deal with the challenges of Network Security.

Background

In July of 2002, Next Generation Security Software Ltd. reported the SQL 2000 server vulnerability. Slammer, or Sapphire, as it was called is a memory resident worm that propagates via UDP Port 1434. In response Microsoft released a security bulletin MS02-039 on July 24, 2002 to address the vulnerability (6 months prior to the Slammer Worm being released into the wild). On January 25,

2003 Microsoft released an updated security bulletin following the spread in the wild.

The Slammer worm, which targets Microsoft SQL Servers, is self-propagating malicious code. According to CERT, this vulnerability allows execution of arbitrary code on SQL servers due to a stack buffer overflow in Microsoft SQL Server. "The worm crafts packets of 376-bytes and sends them to randomly chosen IP addresses on port 1434/udp" (CERT). The worm scans for new hosts, but the current version does not contain a payload.

The Cooperative Association for Internet Data Analysis reported that the Sapphire Worm has been the fastest spreading computer worm to date. During its peak, the Sapphire Worm was spreading over the Internet, doubling in size every 8.5 seconds. "Most vulnerable machines were infected within 10 minutes of the worm's release" (Cooperative Association for Internet Data Analysis). This worm is estimated to have infected at least 75,000 hosts and caused canceled flights, interfered with elections and as well as causing ATM failures (Cooperative Association for Internet Data Analysis). With 42% of all Slammer infections occurring in the United States, the US led all other countries in terms of infections. Most companies and organizations did not even realize that they were running Microsoft SQL. I have listed just a few products below that run a version of Microsoft SQL Server or MSDE that could have been affected during the Slammer attack. As you can see from this list, backups and virus protection servers could have been vulnerable to the Slammer worm (Microsoft).

- Microsoft Biztalk Server
- Microsoft Office XP Developer Edition
- Microsoft Project
- Microsoft SharePoint Portal Server
- Microsoft Visio 2000
- Microsoft Visual FoxPro
- Microsoft Visual Studio.NET
- Microsoft .NET Framework SDK
- Compaq Insight Manager
- Crystal Reports Enterprise
- Dell OpenManage
- HP Openview Internet Services Monitor
- McAfee Centralized Virus Admin
- McAfee Epolicy Orchestrator
- Trend Micro Damage Cleanup Server
- Websense Reporter
- Veritas Backup Exec
- WebBoard Conferencing Server

There are many reasons that the Slammer worm spread so fast. I believe that all of the various reasons played a part in the rapid spread of the worm. An argument can be made that Microsoft is to blame for it's poor coding efforts, or that SQL Administrator's are to blame for not patching their systems, or Network Administrator are to blame for not blocking port 1434 at the router / firewall. All of these are valid arguments. The Slammer worm aided itself in the rapid spread by not producing scanning loops in the code. This allowed the Slammer worm along with the other reasons listed above to spread very rapidly and ultimately generated the denial of service throughout the Internet during the initial release of the Slammer worm.

David Litchfield wrote "proof of concept" code to show the vulnerabilities in Microsoft SQL. The author of the Slammer worm probably used this as his or her template to write the Slammer worm. David Litchfield is a security researcher with NGS Software that publishes his "proof of concept" code to the public in order to assist the industry in preparing for such vulnerabilities. The Slammer worm has been linked to a Chinese hacker group named Honker. Although researchers are 100 percent sure that the Honker group wrote the code they are unsure if they actually release it.

It is important to note that the Slammer worm uses a UDP port instead of a TCP port. Using a UDP port instead of a TCP port allowed the Slammer worm to have less overhead and is faster than the TCP, because UDP is a connection-less protocol which means there is no handshaking before, during or after transmission. UDP packets have no data integrity and skip TCP setup steps such as connection establishment, data transfer and connection release. Those packets that fail are discarded. By using UDP, the Slammer worm gained the desired result of Denial of Service.

There are several things to consider when removing the Slammer worm from an infected system. Disabling port 1434 on your router / firewall or turning off network access to your infected systems will protect your systems from re-infection while you patch your systems. You will need to determine if you are trying to patch the SQL Server version or the MSDE version of SQL, because they require different patches. You will also need to determine the current Service Pack you are running on each system. Using Microsoft's SQL Server 2000 Security Tools is the best avenue when trying to clean or locate multiple infected systems. Using these tools may also identify systems you may not have been aware of. The Slammer worm only resides in network packets and processes. The Slammer worm does not infect the system by writing information to the hard drive. A simple reboot will remove the Slammer worm from any infected system, but without applying the appropriate patches and or service packs the system will quickly become re-infected. As I mentioned in the initial stages of this background section the Slammer worm does not carry a payload. The following example is based on the manual process for removing the SQL Slammer worm.

- Set the SQL Server Service to Manual.
- Restart the infected computer
- Patch the infected system with the appropriate version service pack for SQL Server or MSDE
- Set the SQL Server Service to Automatic

If your system has been infected with the Slammer worm the following processes will take place on the infected system. The Slammer worm will begin scanning for other systems to infect on the Internet. This is done by using the GetTickCount() function inside the Win32 API to generate random IP addresses. For every infection attempt a newly generated address is used. The Slammer worm will use UDP port 1434 in order to connect itself to a remote machine for propagation (F-Secure Corporation).

Much has been written about the Slammer Worm, but one of the most detailed write-ups can be found at http://www.wired.com/wired/archive/11.07/slammer.html.

Countermeasures and Layered Security Approach

Despite this being the quickest worm spreading to date, a number of common security practices could have prevented or mitigated the spread of this worm. Some of these countermeasures can be used by themselves, but together they provide a much more robust layered defense not only effective against the Slammer but potentially against future threats as well. In fact, any port not absolutely needed for public access needs to be blocked at the parameter firewall. When ports need to be open to the public, they should be only on computers in DMZ's, which limit the potential for internal compromise. Port 1433 is used for SQL admin purposes and should not be available to the public.

Blocking UDP port 1434 at the gateway router would have been one way to prevent the worm from infecting vulnerable machines. A Cisco router command would look similar to the following command:
access-list 101 deny udp any any eq 1434 log-input
access-list 101 permit ip any any

Blocking UDP port 1434 on the perimeter firewall would have been another way to prevent the worm from infecting vulnerable machines. A Cisco Pix Firewall command would look similar to the following command:
outbound 2 deny 0.0.0.0 0.0.0.0 0 tcp
outbound 2 deny 0.0.0.0 0.0.0.0 0 udp
outbound 2 permit 0.0.0.0 0.0.0.0 443 tcp
outbound 2 permit 0.0.0.0 0.0.0.0 53 tcp
outbound 2 permit 0.0.0.0 0.0.0.0 53 udp

outbound 2 permit 0.0.0.0 0.0.0.0 82 tcp
outbound 2 permit 0.0.0.0 0.0.0.0 81 tcp
outbound 2 permit 0.0.0.0 0.0.0.0 80 tcp
apply (inside) 2 outgoing_src

Using ingress and egress filtering according to Cisco Systems White Paper Safe SQL Slammer Worm Attack Mitigation would be the most effective way to contain the Slammer worm. I have included the definition of ingress and egress filtering below as well as a default configuration from the same Cisco Systems white paper.

"*Ingress filtering* is typically performed by access control on the perimeter of the network. It is used to block access to hosts and services that should not be publicly available. For instance, it is a security best practice to disallow incoming connection requests to hosts or networking devices unless those hosts or devices are actively participating in providing a publicly accessible service (Cisco Systems White Paper Safe SQL Slammer Worm Attack Mitigation)".

"*Egress filtering* is also typically performed by access control on the perimeter of the network. This filtering blocks a local host's access outbound from the network. Devices that don't need outbound Internet access, such as most of the networking devices in the network or SQL servers that serve only the internal environment, should not be allowed to initiate outbound connections (Cisco Systems White Paper Safe SQL Slammer Worm Attach Mitigation)".

By added both of these filters incoming Slammer packets would be blocked via the Ingress filter and egress filtering would have protected the external network from launching DDoS attacks against other networks (Cisco Systems White Paper Safe SQL Slammer Worm Attach Mitigation). This would allow the Administrator time to detect and or clean vulnerable or infected systems. According to Cisco Systems a sample ACL command for the Slammer worm would look similar to the one below.
Access-list 101 deny udp any any eq 1434
Access-list 101 permit ip any any
Class-map match-all slammer_worm
Match access-group 101
Match packet length min 404 max 404
Policy-map drop-slammer-worm
Class slammer_worm
Police 1000000 31250 31250 conform_action dropvilate-action drop

Using an internal host based Intrusion Detection System on all SQL servers; configured to allow only SQL Administrators (Internal Static NAT IP addresses) to connect would have reduced the spread of this worm. In addition, it is likely that a host-based Intrusion Detection System would have detected and blocked the spread of the worm without explicate policies to allow such traffic. One such

Hosted Based Intrusion Detection System, BlackIce came out with a new pattern file or detection file for the Slammer worm in version 3.6cdb. However, properly configured host base Intrusion Detection Systems would have blocked the incoming and outgoing connections needed to propagate the worm assuming that it had not been explicitly permit for UDP 1433 to allow traffic.

Many Network Based Intrusion Detection Systems have the ability to block traffic based off statistical anomaly as well as signatures. Having a Network Based Intrusion Detection System in place would have likely helped reduce the infection and spread of the Slammer Worm.

Although this next layer of security would not have helped during the initial release of the Slammer worm, it is worth mentioning as a vital piece of a layered security approach. Currently, my organization uses a multi-layered approach in regards to our virus protection. We are using Trend Micro's Viruswall (1st layer) at the gateway, Exchange ScanMail on our email server (2nd layer), ServerProtect (3rd layer) on all Windows 2000 Servers and OfficeScan on Desktops and Laptops (4th layer). We also block the following attachment types at the gateway (Ade Adp Bas Chm Cmd Cpl Crt Hlp Hta Inf Ins Isp Lnk Mde Msc Msi Msp Mst Pcd Reg Sct Shb Vb Vbe Wse Wsf Shs Vbs Exe Com Mp3 Pif Bat Wav Scr Att Eml Nws Dll). When a Virus, Worm or Trojan is detected by one of these products it is automatically deleted and a Systems Administrator is alerted of the deletion. We do not use the quarantine option within any of our anti-virus products. We have all of our anti-virus products set to check for a new virus pattern file each hour. If a new pattern file is available it is automatically downloaded and pushed to our clients and servers. The Systems Administrators are also notified of the new pattern file as well. If a client or server is turned off for any reason during the push process the new pattern file will be loaded via logon script once the next user logs onto the system. This approach has been very successful and has blocked thousands of Viruses, Worms and Trojans over the past three years.

Again, this next layer of security would not have helped during the release of the Slammer worm, but it is worth mentioning as a vital piece of a layered security approach. It is also one layer of defense that is most commonly overlooked in the overall scheme of Network Security. Wireless network installations by-pass many if not all layered security and countermeasures. When designing a wireless infrastructure today you should consider a vendor who provides the best available security out of the box. My organization has implemented a Cisco Wireless Infrastructure using 1200 series wireless access points. We have set a SSID that does not disclose any relevant information to our organization. We also use dynamic WEP (LEAP, EAP) keys that change every 15 minutes and users are validated via Cisco ACS. Using a static SSID and WEP key can be used as backdoors into your network. Cisco has other wireless security hashing algorithms built into their IOS. Cisco's TKIP and MIC are other enhancements that should be used to protect your wireless infrastructure. My organization has

also implemented a wireless IDS from Air Defense to help assist with protecting our wireless infrastructure. The Air Defense IDS looks for rouge access points and clients. Once a rouge access point or client is detected the Systems Administrator's are alerted. We also scan our wireless network with Air Magnet and in the past have used Network Stumbler to ensure our existing access points are set to our internal wireless standards. If your organization cannot afford a wireless IDS, I would suggest using a filter by MAC address approach on each wireless access point. Depending on how large your wireless user base is this option may have an impact on performance.

My organization has implemented port security on our internal Cisco switches. This has allowed us to detect when a laptop or any other Ethernet device has been plugged into our network without our knowledge. Port security can be set to allow only one MAC address to function per port. For example, if a consultant comes in to assist with a project and plugs his or her laptop into our network the port will shut itself down. Since, the consultant's laptop has not been checked for Viruses, Trojans or Worms we have no idea what kind of damage he or she could have caused while on our network. Once the consultant realizes they do not have connectivity the Information Systems Department will be contacted to assist. At that time the Information Systems Department can ensure the laptop has the latest Microsoft patches, virus software and check for any other destructive software. I have included a brief description from Cisco about how they define port security below. "You can use port security to block input to an Ethernet, Fast Ethernet, or Gigabit Ethernet port when the Media Access Control (MAC) address of the station attempting to access the port is different from any of the MAC addresses specified for that port. The global resource for the system is 1024 MAC addresses. In addition to this global resource space, there is space for one default MAC address per port to be secured. The total number of MAC addresses that can be specified per port is limited to the global resource of 1024 plus one default MAC address. The total number of MAC addresses on any port cannot exceed 1025" (Cisco Systems).

The most important step that should have been taken to prevent the worm would have been to patch all vulnerable systems immediate following the availability of Microsoft's patch. This would have addressed the root issue and removed the vulnerability rather then masking it. While this is good advice, even Microsoft has a tough time following this recommendation. On January 28, 2003 UK Technology news, reviews and downloads (www.vnunet.com) reported that Microsoft didn't patch all of their SQL servers and were attacked by the Slammer worm as well.

Microsoft did release a patch in July of 2002 (MS-039) for the SQL Slammer vulnerability. Microsoft released a patch in October of 2002 (MS-061) for Elevation of Privilege in SQL Server Web Tasks with included the SQL Slammer vulnerability. Microsoft also recommended that Administrators load SQL Service Pack 3 as well; which included all the latest hot fixes, patches as well as the SQL

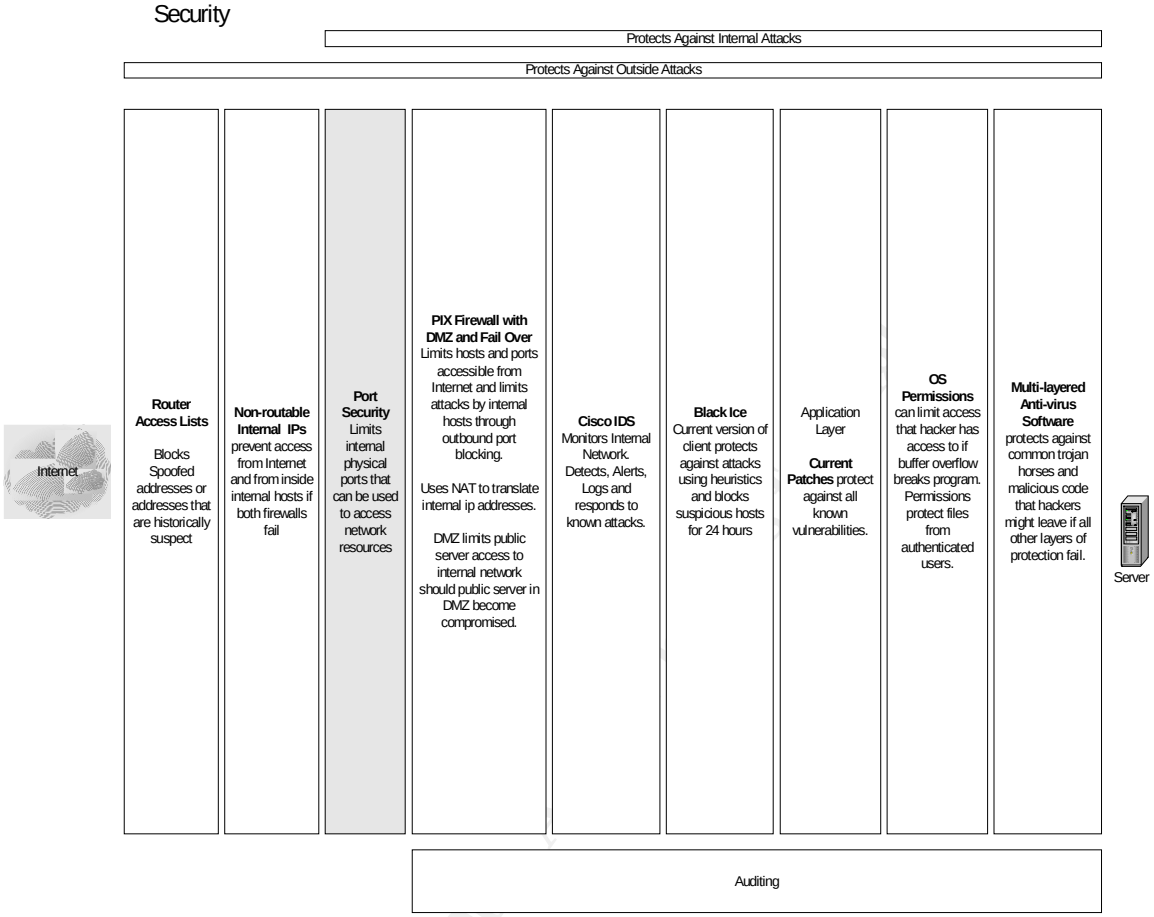
Slammer vulnerability. Microsoft also created several utilities: SQL Server 2000 SQL Scan, SQL Check and SQL Critical Update for removal and detection of the Slammer worm.

Writing internal policies as well as procedures is probably the least favorite duty for all IT people to perform. It is also one of the most important. Having internal written policies will assist in the process of alerting and educating your staff and organization before issues arise. In some cases your written internal policies maybe your only means of enforcement. I have included the titles of a few my organizations internal polices, that I believe are a must for any organization.

- Authorized Users
- Computer Network and Internet Use
- E-mail Use
- End of Service
- Software Use and Duplication
- Network Passwords
- Computer Anti-Virus Software
- Security Violation
- Remote Access

The following Visio diagram below shows my organizations outline to a multi-layered security approach.

Internal Public Server Layered Security



My day during the Slammer Worm

On Saturday January 25, 2003 the media was a buzz with the latest worm to sweep through the Internet. Thousands of computers were affected including retail stores and even Microsoft itself. Organizations around the globe were working that weekend to try and mitigate the spread of the worm and even though the worm was not destructive the denial of service that resulted, the downtime and resources required to respond had a lasting affect as organizations learned a difficult lesson in network security once again. I played golf that day confident that the proactive security counter measures, business continuity and disaster recovery testing my organization had in place, we would be prepared to withstand just such an attack. I did receive a call from our weekend on-line weekend support staff, stating that we had lost Internet capabilities. Due to the packet flood caused by the Slammer worm our ISP was forced to shutdown connectivity. Having our Internet access disabled by our ISP could have caused some additional problems if our systems had not be properly patched and would have needed Internet access to download patches. We are a

Microsoft Premier Client and thankfully had all the latest service packs on CD on-site. We regained Internet activity around 8:00 PM EST the same day. We owe much of our education and success to the lessons learned during and after 9/11. I knew with the steps taken by my staff and myself had eliminated the threat that this vulnerability posed months before hand. In this paper, I discussed the layered defense architecture that we have in place that has not only successfully protected my organization from the Slammer Worm but other worms and viruses. This is not theory. These countermeasures have protected my organization from Slammer, Code Red, Nimda and many other worms and viruses. We have developed a security team that meets weekly and is made up of Network Administrator's, Applications Programmers, Client Services Analysts as well as Management. This has allowed us to gain different perspectives on security and how it affects different areas and groups. Sharing our security architecture within the Information Systems Department has allowed us to use a top-down approach in getting everyone on-board. These security meeting has also allowed use to follow-up with a Change Management process (Patch Management) that everyone involved can agree on system downtime when a patch or change is required. We also have quarterly penetration test performed by TruSecure Corporation as well as another third party company to ensure confidentiality, integrity and availability. We also have two or three Disaster Recovery test annually. These test have occurred in different locations and at different times within the United States multiple times. We send some our Disaster Recovery team members by plane and some by car. Backup tapes as well as documentation and procedures are stored off-site daily and are shipped separately, days before the Disaster Recovery test is scheduled. We have multiple employees within multiple departments inside and outside of Information Systems Division that can declare a disaster by calling our Disaster Recovery vendor 24 hours a day 365 days a year. Within the next few months we will be performing a disaster recovery test without anyone's knowledge inside of the Information Systems Division besides the Director. We have proven that we can recover when we know a test is going to be performed, this time we will have to prove that we can do the same without the knowledge of when. The Director of Information Systems will call all of us into a conference room and say what the disaster is and who is available to perform the recovery. It will be up to the people available to follow our procedures in order to gain transportation, lodging and see that the tapes are on their way to the disaster recovery site. The newly assembled disaster recovery team will also need to declare the disaster by contacting our disaster recovery vendor to ensure they have adequate space and equipment available at our preferred disaster recovery site. If our disaster recovery vendor cannot support our needs at our preferred site we will need to go to our alternate site. We also perform Business Continuity testing on-site. We have multiple computer rooms that have the exact equipment stored in separate buildings. We have test that allow us to simulate our fail over capabilities in case a key production server crashed by loaded a patch or perhaps a hardware failure. In March of 2003 my company received Certification from TruSecure Corporation for our security efforts. This does not mean we can

now sit back and not worry about security related threats. We must continue to be diligent with patching our systems as well as monitoring our current countermeasures to ensure operability. Business Continuity, Security and Disaster Recovery all work together to ensure our confidentiality, integrity and availability.

Conclusions

While many organizations have learned hard lessons in the past and have devoted the necessary time and resources into implementing network security best practices and countermeasures, it is clear many have not. Although many companies and or organizations may not be able to afford experienced Administrators that have been taught how to develop a layered approach to security for their company or organization. Companies or Organizations with Administrators that have the knowledge or experience often have a difficult time convincing management to allocate the necessary funding to secure the assets for which they are trying to protect. It is very clear that even the most experienced Administrator's are not patching their systems. The most common reason is downtime. Each company or organization must develop a patching process that includes testing first, and follow it religiously. This is the most important and most overlooked countermeasure in a layered security approach. One U.S. Senator has written legislation that will require Businesses or State Government to notify individuals when a database has been compromised that had or has personal data in it (U.S. Senator Dianne Feinstein). This law includes Social Security numbers, Driver's numbers as well as credit card numbers and became effective July 1, 2003 (U.S. Senator Dianne Feinstein). With laws like this one being passed upper management may soon be forced realize the importance of security and the need for a layered security approach to protect their organizations confidentiality, integrity and availability. With an estimate of over 75,000 Slammer infections worldwide, many organization's still fail to realize the importance of a layered approach to security and the need to patch their systems. SQL Slammer spread across the world in 8 minutes, the next worm maybe even faster. Organizations and Administrators must take this threat seriously in order to protect their organizations confidentiality, integrity and availability. Not only does this make good business sense, it is the responsibly of Netizens to be a good neighbor and to take steps to ensure network resources are not misused in ways that may negativity affect others on the Internet. It is anyone's guess, which unprotected UDP port, will be exploited next in order to release the next Warhol worm attack in the wild. Implementing the basic countermeasures will go a long way in preventing the Slammer worm as well as future worms.

References

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/virus/alerts/slammer.asp

http://www.microsoft.com/security/slammer.asp

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS02-039.asp

http://www.cert.org/advisories/CA-2003-04.html

http://www.caida.org/outreach/papers/2003/sapphire/sapphire.html

http://www.f-secure.com/v-descs/mssqlm.shtml

http://www3.ca.com/solutions/collateral.asp?CT=27081&CID=39147

http://www.cnn.com/2003/TECH/internet/01/26/internet.attack/index.html

http://www.sophos.com/virusinfo/articles/slowinternet.html

http://vil.mcafee.com/dispVirus.asp?virus_k=99992

http://www.trendmicro.com/NR/rdonlyres/ei5fy2w7c3t5lnhg6qbbla2rifexns3nysk2qt6u4cngm2zllqpmznfrb4bouf727iih44bt5ebauf/slammer3.pdf

http://securityresponse1.symantec.com/sarc/sarc.nsf/html/w32.sqlexp.worm.html

http://blackice.iss.net/update_center/readme_pcp.txt

http://www.cisco.com/en/US/netsol/ns110/ns170/ns171/ns128/networking_solutions_white_paper09186a008013573f.shtml

http://www.eeye.com/html/Research/Flash/AL20030125.html

http://www.internettrafficreport.com/event/3.htm

http://www.vnunet.com/News/1138312

http://www.wired.com/wired/archive/11.07/slammer.html

http://www.cisco.com/en/US/products/hw/switches/ps679/products_configuration_guide_chapter09186a008007ef1a.html - 18398

http://www.senate.gov/~feinstein/03Releases/datasecurityrelease.htm

http://www.airmagnet.com/

http://www.airdefense.net/

http://www.theage.com.au/articles/2003/02/07/1044498960818.html

http://www.zdnet.com.au/newstech/security/story/0,2000048600,20271839,00.htm

http://www.dynamicnet.net/news/articles/slammer.html

http://www.eweek.com/article2/0,3959,848899,00.asp