Global Information Assurance Certification Paper

Small-site Information Security on a (very loose) shoestring – a case study.

Michael P. Millow, CISSP
January 4, 2003
GIAC Security Essentials Certification Practical Assignment
Version 1.4b (amended August 29, 2002) – Option 2

Abstract
Large corporations recognize the need to invest manpower, time, and money managing their system and network infrastructures. Most of these companies have also recognized the value in focusing specifically on information security to protect and manage their assets, secrets and reputations. Unfortunately, this same understanding of the need and value of information security is not seen at a significant portion of midsize and smaller companies. This may be because of the perceived cost and/or complexity, management attitudes, or simply a lack of knowledge. This lack of understanding puts all Internet users at increased risk of attack or compromise.

This paper will describe one such smaller company and the state I found it in when I joined it. This will be followed by a review of corrective actions (and their limitations) that significantly enhanced the overall security posture. This was accomplished while working with management attitudes that did not generally hold information security at a high value in the day-to-day activities of the company. Corrective work was done over a period of about a year, by the end of which many improvements had been realized. More importantly, the management team at this site was much better educated in the value of information security and had become willing to invest some limited resources in security activities.

Overview
When I started with this company they did not have a dedicated Network & Systems Administrator at the site, nor did they have personnel focused on information security. This was at a site that, at the peak of employment, had nearly 70 people working in it. This site hosted over 200 network-connected devices, was connected to the Internet, and had email, ftp servers, several active databases and several source-code tree structures.

Site management held a "the engineers can do it" attitude around maintaining the site infrastructure and an even lower apparent priority for security related activities. Fortunately, after presenting the data found in my initial assessment of the site, it was a relatively quick sell to get management to understand that while the engineers could technically "do it" - and keep things running - they were being paid to do other things and there was sufficient value in hiring dedicated network and system administration and information security staff.

This site had been in existence for several years, and had absorbed the systems and culture from two different groups of engineers without really consolidating the differing environments. The site was really two sets of parallel systems, services and procedures running in the same building. In addition, since the maintenance of these systems had been a secondary task, it had been spotty at best. There had never been dedicated personnel to manage infrastructure issues as the building evolved, leaving a patchwork of network and backbone services. Finally, as this had all been ad-hoc work by personnel who's primary tasks were unrelated to security and administration; documentation for the environment, policies and procedures was non-existent.

My plan of attack to bring this environment under control, and provide the maximum security given the constraints included the following phases:

1. An initial assessment of systems, infrastructure, policies and procedures including a review of what vulnerabilities and risks exist
2. Define the critical systems and resources (required management input)
3. Determine corrective actions to be taken to protect the critical systems and resources, including actions enhancing a defense in depth posture
4. Complete these corrective actions
5. Evaluate the impact of the corrective actions and communicate new current state to management
6. Return to Step 2 and repeat

Initial Assessment

The initial state of this site had several bright spots, but generally left me with a period of sleepless nights until some of the corrective actions could be finished. A review of what I found follows, broken into three areas – policy, infrastructure and systems.

<u>Policy</u>

There were no security policies in place at this site (or company wide). Areas that needed to be covered included an overall security policy, anti-virus software, passwords, account management, email and ftp usage, peer-to-peer software usage, DNS change control and incident handling.

There were also no procedural documents when I first started. Some of the documents that were needed included configuration instructions for email, ftp, DNS, NIS+, the backup system, the anti-virus server, the firewall, and database management. Procedural documents pertaining to the policies listed above also were needed.

<u>Infrastructure</u>

The infrastructure was a patchwork of loosely connected systems; "highlights" of areas that increased the security risks of the site are listed:

- A stateful inspection firewall was in place without a dmz segment
- Multiple DNS domains (internet accessible, hosted inside firewall)

- Multiple FTP servers (internet accessible, inside firewall)
- Sendmail and Exchange 5.5 servers – internet accessible, inside firewall)
- 1 customer support database, 2 bug tracking db (1 linked through firewall)
- File systems: NFS, automount, samba, WinNT domain, NetBIOS shares
- NIS+ and Windows NT Domain not linked together, duplicate user accts
- Out of date account management (active 'dead' accounts, orphaned data)
- Solstice Networker backup covering major servers, but not email
- There was no IDS (intrusion detection system) in place
- Unknown number of keys to wiring and server closet
- No auxiliary cooling or power management for server closet
- Poorly managed physical network, unrelated services and systems sharing older hubs allow traffic sniffing

Systems
There were numerous versions of server operating systems running meaning patch management would be difficult as each server required unique patch versions; more "highlights" are listed here:

- Multiple Server OS's (5 Solaris versions, 3 Windows versions and Linux)
- No security patch mechanisms in place
- No logging on any systems (other than simple firewall logging)
- Two incompatible source code control mechanisms
- No site-wide anti-virus software in use
- Analog KVM switches (required attended reboots after power outages)

The site was found to be a confusing collection of loosely related or duplicated systems connected by a poorly configured network infrastructure. Keeping this diverse environment current for security patches would be very difficult. Services that should have been isolated from the internal network were running on the same network segment as the more sensitive databases and code trees. Policies and user understanding of proper procedures were non-existent, or at the least not documented. Basic protections such as anti-virus software and a DMZ were missing. There was much work to be done.

Define the Critical Systems
The findings of my initial assessment had been shared with the site management in an effort to justify the creation of a fulltime MIS/security position as described earlier. After seeing the assessment, they quickly understood that there were risks present at the site.

They were able to see risks both from a variety of external factors such as incoming viruses and intrusions, and from internal factors such as failing backups or lack of controls on accounts for terminated users. They identified the critical systems and services the site needed to function, but in the end were still unwilling to devote much money or resources other than my salary to corrective actions. This meant I would be working under the radar and on a shoestring to accomplish my tasks We ended up working with the following list of critical areas I was to focus on protecting:

Data structures
- Product source code
- Bug control and Customer Support Databases
- Email

Services required
- Internet access
- Site-wide anti-virus protection
- FTP, Email and DNS services
- Daily backup of all critical servers/data

Corrective Actions
After having management identify which systems and services they considered critical, I set about identifying the vulnerabilities which affected these systems and planning corrective actions. I also looked at other activities I saw which had a high payoff/low cost and could be easily incorporated into the corrective actions list. Most of the items I added to the list focused on policy and procedure, which you will notice was largely absent from their list of critical areas. Throughout this exercise, I focused on maximizing the impacts of these changes by working towards defense in depth. “Defense in depth is the concept of protecting a computer network with a series of defensive mechanisms such that if one mechanism fails, another will already be in place to thwart an attack.”[1] (McGuiness).

The time period to complete this first round of corrective actions was in fact quite long. Close to one year was spent planning and implementing these changes. There were periodic status updates during this period, as well as periodic priority adjustments within this list of actions as well as in balancing these activities with the day-to-day administrator activities needed to keep the site running. The first round of corrective actions that were completed are as follows:

- Create a security policy
- Create and implement a site wide anti-virus policy
- Create and implement a password policy
- Create and implement a security patch management policy
- Create and implement a user account policy
- Create and implement a key policy for the server room
- Create and implement a backup policy
- Close Win9x network shares
- Setup a DMZ
- Consolidate to a single email system
- Replace/rewire hubs to consolidate like traffic

- Install KVM switches
- Install Big Brother for system and network monitoring
- Document network/server configurations

Create a security policy

An overall security policy has still not been created for this site. This is due to the lack of management willingness to devote the resources to such an exercise. Taking a piecemeal approach to documenting and formalizing the security posture at the site is eventually bringing management to accept the idea of creating and abiding by policies. This is being accomplished by creating topic-specific policies and procedures that management can accept and support one at a time rather than one large over-arching structure. The end result is that the site is protected by a series of separately defined defenses instead of one comprehensive policy. While it is frustrating at times to not have management support for general security goals, the work is still being accomplished (and the protections gained) by presenting the process in smaller bites to be digested and accepted. This may not be the "best" way to achieve site security, but it is much better than the earlier lack of concern seen at this site.

Create and implement a site wide anti-virus policy

I chose McAfee anti-virus software configured to use a server based push method to keep dat files current on the clients. The anti-virus policy included mandatory anti-virus software on every desktop in the building and required users to allow the regularly scheduled updates and local drive scans to occur. The policy also required that anti-virus software be run on the mail server, and limited the types of attachments the mail server would pass to users. Finally, the policy specified that all file servers run anti-virus software as well and that they perform periodic checks of files stored in user home directories and on other server directories. Unfortunately, this strict policy was only site wide not corporate wide, the home office chose to implement a much looser policy. We continued to receive infected emails from employees at the parent office, making the filters running on the mail server that much more important.

Create and implement a password policy

A password policy requiring minimum standards for passwords was created. A strong password was defined based on the SANS/FBI Top 20 List[2] recommendations. The policy specified that passwords not be dictionary words, have 6 or more characters, with one or more number and one or more special character, that users change their passwords on a regular basis, that they memorize the passwords and not leave them written in their work area, and that they not share the passwords with others.

Both of the user account management systems (NIS+ and NT Domain) had capabilities to enforce parts of this policy using password aging, minimum length and complexity checking. Management decided that forcing users to comply using the operating systems was too drastic, so education and reminders were the only means used to enforce the policy. Having any password policy was a significant improvement in that it communicated to users what was expected of them to help keep the network secure. Mandatory enforcement could have further enhanced security, but was not implemented at this site.

<u>Create and implement a security patch management policy</u>
This site hosted a wide variety of server operating systems and infrastructure services, all of which suffer from periodic vulnerabilities. These devices require patches be installed to maintain a secure posture if the site is to safely remain connected to the Internet. As was noted earlier, patch management was not an area that received a great deal of attention prior to my taking on the task.

This process included several parts:

1. Assess the initial state of all systems (both OS and applications)
2. Obtain, test, document and apply all needed patches for each system
3. Regular monitoring of ongoing patch releases, repeat step 2 as needed
4. Write a security patch management policy and procedures

The observant reader will note that the last step above should really be the first thing done, but given the significant exposures with these systems, it was decided that the more important first task was to bring the systems up to date, and that the formal policy and procedures could be documented after the systems were better protected. This reversal of tasks, where the vulnerabilities were closed and then policies and procedures were written happened in several areas during this period.

<u>Create and implement a user account policy</u>
First I needed to identify which accounts were no longer in use. These dead accounts were then disabled for a period of one month to ensure no process breakages. This step was taken due to the lack of documentation and uncertainty around what these accounts had access to or were used by. During this holding period management reviewed the directories related to each account to ensure no critical software or data would be lost. Certain directories were backed up to CD for storage “just in case”. At the end of this waiting period the accounts and directory structures were purged.

As this process was being completed a policy was created along with specific procedures for who was responsible for alerting the administrator of employees leaving and how to review and archive relevant data structures and remove the accounts and directory trees.

Cleaning up these dead user accounts removed the opportunity for disgruntled former employees to access the network using their old accounts, current employees to disguise their actions by accessing resources using these old accounts and outsiders from hacking into the system using accounts that weren’t being monitored regularly.

Create and implement a key policy for the server room
During the initial review, it was determined that no one knew who had keys to the server room, or how many there were. This room hosted our infrastructure servers as well as the wiring racks, network infrastructure and telephone switch. Management approved a list of who needed access and a policy was created where each authorized employee had to sign for a key and agree to terms of use. The door was re-keyed and the authorized users were issued new keys.

This simple (and inexpensive) change dramatically increased the physical security of the network infrastructure by preventing unauthorized access for malicious intent as well as unauthorized configuration changes which may have had unintended consequences later.

Create and implement a backup policy
Most of the critical systems were already being backed up using Solstice Networker software. There were several shortfalls in with the backups however:

- Backup and restore procedures were not documented – the one person who knew the system left the company six months after I joined
- The data structures on several servers had changed and the corresponding changes were not made in the backup configurations
- The Exchange mail server was not being backed up
- The restore function had not been validated
- Monthly backups for offsite storage would sit on the assigned managers desk for weeks before being moved offsite

A backup policy was written, along with detailed procedures for backing up and restoring all systems as well as configuring and managing the backup system. Additional software was installed to allow the system to backup the mail server and configuration changes were made to reflect current directory trees. Several restore tests were performed. Finally, the manager assigned storage duties for the offsite tapes received frequent reminders to take them offsite until he adopted better habits.

These changes paid off several times when drive failures required rebuilding several systems from the stored tapes, in each case the restore procedures successfully recreated the missing data.

Close Win9x network shares
While the firewall was blocking the incoming NetBIOS ports, open network shares on Win9x clients still presented a security risk on the internal network segment. Attacks such as SirCam[3] and Nimda[4] both spread quickly through a network segment over these unprotected shares once they find their way into the network.

One solution to these unprotected shares would be to upgrade all older Windows platforms. That was not possible at this site as these clients were used for regression testing by several departments. User education and periodic review of the systems ensured that these machines had their file and print sharing disabled, removing this vulnerability from the network.

Setup a DMZ

The site had a stateful-inspection firewall in place when I joined the company. It was configured to allow all needed services to pass directly to the internal network segment as well as to allow any outgoing connections. This exposed the internal network to a variety of vulnerabilities. The first task was to identify all open ports into the network and determine if there was real business need to justify them. The firewall had been fairly carefully configured and it was found that no open ports could be closed without impact, several could be limited however to be accessible only from the companies other sites rather than the entire Internet and those changes were made.

More significantly, a DMZ was created and several services were either moved to the DMZ outright (ftp servers and demo servers for our products), or had front ends put in the DMZ that redirected queries into the required services (email and a bug database shared with external partners). Moving these services into a DMZ significantly reduced the vulnerability of the internal network by removing these Internet accessible services from the private segment.

Consolidate to a single email system

The site originally was running parallel email systems, Exchange 5.5, and sendmail on an older Sun box. The site was transitioned to use Exchange only, which required moving about 30 user's mailboxes and addresses to the Exchange server. Removing the sendmail server from the network (and closing the hole in the firewall to that service) eliminated exposure to several sendmail vulnerabilities and removed one software package and one older server running a unique OS from the list of things that needed to have patch maintenance implemented.

I further reduced vulnerability by creating an email proxy in the DMZ to separate the internal Exchange server from the Internet. This reduced the chances of our email server being compromised, and created another layer of defense for both filtering traffic and limiting access to an internal system.

Replace/rewire hubs to consolidate like traffic

The wiring rack for the building was rebuilt, including replacing a number of older hubs with newer, faster switching hubs. Concurrent with the rack rebuild, the wiring schema was reworked to group systems used for similar tasks together on the same hubs/switches. This dramatically improved the overall network performance and greatly simplified management of the network backbone when the occasional problem arose.

More importantly, it removed a security hole by preventing unauthorized sniffing of network traffic. Under the old wiring scheme the finance server may have been on the same hub as development engineers, or the compile server on the same hub as a marketing manager. Since these were older hubs, sniffing the traffic on each hub was possible from any other device hooked to that hub. By both grouping 'like' traffic and migrating to switched hubs, the amount of traffic any one device could see was dramatically limited - in many cases, individual devices could only see their own traffic after the rebuild.

<u>Install KVM switches</u>
Most servers in the server room shared keyboards, video and mice (KVM) due to heat and power constraints. Several of these switches were old style analog knob switches. The building suffered from periodic power outages, and restarting the servers on these analog switches was a manual process. A new KVM switch was installed which allowed for unattended restarts on all critical servers, this reduced the chances that a power outage would become a denial of service event when a system became unavailable due to a failed restart. This allowed much greater uptimes for resources accessed by business partners in other parts of the world who didn't share our time schedules.

<u>Install Big Brother for system and network monitoring</u>
This system had a positive impact on the overall security of the network by adding another layer of defense in depth. Using this system, critical services could be monitored on a per-server basis and email or pager alerts could be forwarded when needed. Also, outages of a service for more than just a few seconds were logged and would display on a status console for later monitoring[5].

This accomplished two things; real-time notification of outages that may be related to compromised systems or DOS attacks, and later review of unscheduled outages to determine cause. Reviewing the status console after a period of unattended operation (last night or over the weekend) could be done in a few seconds by checking a status screen showing colors – green shows no changes, purple shows a previous outage that has been restored. This allowed me to instantly see if a service had stopped and restarted over night or the weekend – I could then check that system more closely to determine why the restart had occurred, hopefully catching any compromises.

<u>Document network/server configurations</u>
A network binder was created and stored in the server room at this site. This binder contained the following sections:

- Written description of each server: platform, physical disks, memory, OS, revision level, directory structure, mount structure (where appropriate), domain, services/processes (including specific data where appropriate), backup method, patch history and maintenance history

- Schematic describing the premise wiring for both voice and data, including a wiring matrix for the rack and a switch/wall port/device matrix
- Copies of the various protocols and procedures needed to perform administrative tasks
- Contact sheet for ISP providers, facilities managers and critical vendors
- Security patch routines for systems and services such as web/email/ftp including where to obtain/verify and how to test/install/document patches
- Sealed envelope with up-to-date master password list and sign-in sheet

Creating this binder served a variety of needs, including documenting how to perform a wide variety of procedures for future personnel, documenting the current state and history of the systems, and providing guidance for recovery processes by clearly documenting the known state of systems. This exercise also pointed to further work needing to be done in comprehending, securing and managing the environment based on holes found in the knowledge collected.

Evaluating the Impact

This evaluation consisted of two components:

- Assess the impacts of the completed actions
- Identify items in the initial assessment that had not been fully addressed and identify any new risks and vulnerabilities in the environment

<u>Impacts of the completed corrective actions</u>

Completing these corrective actions created significant improvements in the sites security posture. The initial assessment and the critical resources list had been done using qualitative measures that had not assigned dollar values to the risks and vulnerabilities or to the resources that needed to be protected. Following that lead, the impact evaluation was also completed and presented using a qualitative model. In hindsight, a quantitative approach, linking the exposures and the resources to dollar values, may have allowed management to more fully understand the value of the security work being undertaken.

Management did see value in the corrective actions, including:

- Only two virus incidents at the site in a year, with those limited to one and three machines respectively – in comparison our parent office suffered over a half a dozen incidents with dozens of machines infected.
- No measurable intrusions into the network in a year through the firewall. Firewall logs did show numerous superficial scans and several concerted scans of our network over the year.
- The server platforms received numerous patches and software/security updates. In some cases 20+ patches were applied to a single server. The application of these patches significantly reduced the vulnerability of these systems to known problems and attacks.
- Documentation of the network, systems, policies and procedures was created. The network binder supports current and future employees in managing the environment and responding to incidents more effectively.

- Formalizing of various policies, while originally resisted by management, was a positive move as it increased the administrators, managements, and the users awareness of what was expected of them as well as knowledge of standard methods to perform a variety of tasks.
- Old user data was archived, possibly saving source code and tools, which could have been lost as old directories were purged from the system. The company had experienced a significant turnover in both engineers and their first level managers, so knowledge of what was where had been lost. The potential loss of important pieces of information had been mitigated.
- Reworking the network wiring increased diagnostic speeds when issues occur due to improved configuration and documentation. Additionally, the risk of unauthorized sniffing of network traffic was greatly reduced.
- Removing the sendmail server from the network eliminated exposure to several vulnerabilities and removed a software package and server OS from the list of patch maintenance items.
- Adding a key policy and re-keying the server room dramatically increased the physical security of the network infrastructure.
- Updating the backup system, documenting procedures and testing the restore functionality paid off several times during the year when production systems threw drives and needed to be rebuilt from the tapes.
- The management team is more comfortable that resources need to be invested to maintain a good security footing. While there is still a desire to do things "for free", they are now willing to engage in discussions of potential risks and provide more support for keeping the site secure.

<u>Continuing issues</u>
There are still challenges at this site. Several significant changes need to be made in the next round of corrective actions based on the initial assessment performed over a year ago.

- Site wide logging is the next item to be implemented. One of the Solaris systems will host the log files, with all others copying their alerts to this one system. A log-monitoring package such as Swatch[6] will be added to ease the burden of reviewing the logs and increase the chances that improper activity will be noticed in a timely fashion.
- Convincing management of the need for a site-wide security policy remains. The benefits gained so far from the target-specific policies could both appear to justify the gains a formal policy would produce and appear to justify that such a policy is not needed, as the details already exist in the topic-specific documents. Further work needs to be done in educating management on the benefits of a formal, adopted security policy.
- Creation of an incident handling policy and related procedures and resources should be done. This will allow the site to respond according to predetermined guidelines to security incidents, which will minimize the disruption of those incidents and limit the damage they may do.
- There is still no IDS system in place. Since there won't be much money available to implement this, a Linux server will be placed on the network

segment between the perimeter router and firewall to monitor incoming and outgoing traffic. A specific IDS package needs to be selected for this system. Adding an IDS will significantly increase the defense in depth of the network by watching for early signs of incoming attacks as well as signs of internal infections trying to get out.

- There is still no auxiliary cooling or power management for server closet, leaving the infrastructure vulnerable to power outages as well as system failures due to overheating during certain times of the year. Both of these items require significant funds to correct. Given a lack of money, these issues will continue to be compensated for by ensuring systems can auto-restart after a power outage and maintaining proper backups for rebuilds.

Conclusions

Security work is never done, monitoring new patches, changing conditions and evolving threats will continue. Many of the corrective actions listed above will still need to be completed. The site continues to change, and so do management priorities.

In my case, management has come around to see value in what a security professional offers to a site, but they still want the work done on a shoestring, and they still don't want disruptions or to have changes interfere with site productivity. It was the intent of this paper to give the reader who may be in a similar situation hope – and some ideas on how to accomplish your security goals while helping management understand their importance at the same time.

1. McGuiness, Todd. "Defense in Depth." SANS Info Sec Reading Room. November 11, 2001. URL: http://www.sans.org/rr/securitybasics/defense.php (8 Jan 2003)
2. SANS/FBI Top 20 List "The Twenty Most Critical Internet Security Vulnerabilities (Updated) ~ The Expert's Consensus" Version 3.21. October 17, 2002. URL: http://www.sans.org/top20/ (8 Jan, 2003)
3. "CERT Advisory CA-2001-22 W/32/Sircam Malicious Code." August 23, 2001. URL: http://www.cert.org/advisories/CA-2001-22.html (8 Jan, 2003)
4. "Cert Advisory CA-2001-26 Nimda Worm." September 25, 2001. URL: http://www.cert.org/advisories/CA-2001-26.html (8 Jan 2003)
5. "Big Brother is watching. Are you available?" URL: http://bb4.com/index.html (8 Jan 2003)
6. Atkins, Todd. "SWATCH: The Simple WATCHer." July 26, 2000. URL: http://www.stanford.edu/~atkins/swatch (8 Jan 2003)