Taking the eGrade Card Challenge: Application to a Small Business eCommerce Site. A Case Study

Brian Bennion

GSEC Practical v1.4c
January 15, 2005

Abstract

This paper describes the deployment of a small business ecommerce website using a previously proposed 'egrade' (1). The current status of the website will be discussed within the framework of the sercurity essentials as well as given an initial grade based on the “eGrade Card”. Deployment of the ecommerce software, security enhancements, and policy creation will described in detail. A second eGrade Card assessment is presented and discussed as it applies to the entities business plan and available resources.

Introduction

Many small businesses are implementing a presence on the world wide web (WWW). Most rely on the Internet for advertising their wares as well as online purchasing of their products. There are several advantages for small businesses pursuing this model (2). Firstly, overhead costs can be minimized. A small business may not need or want an expensive storefront (i.e. Brick and Mortar) and all the maintenance and associated liability. Secondly, the Internet allows for largely anonymous contact with potential customers as well as the associated convenience (i.e. customers sitting at home). This type of contact provides a low pressure environment for the customer. However, the small business must work harder to attract and retain customers gained from an Internet website as a competitors business is just a click away. Furthermore it is harder to make personal contact and form relationships with a customer in cyberspace where body language can not be communicated. These and other disadvantages can be mitigated with an optimal website and constant attention to customer service and satisfaction.

Several large online retailers have initiated ranking/grading systems in order to continually improve contacts/relations with customers (3). Small businesses can implement similar features into their website. One often overlooked detail in online commerce is a proper information security posture. A failure in security/confidentiality will harm a business more than anything else and therefore should be the guiding principle in deploying/implementing an 'ecommerce' Internet site.

This case study will report on the implementation and deployment of a small business ecommerce website. The paper will detail the initial network topology, hardware, software configuration, and policy development. An initial “egrade” assessment will be given and discussed in light of the current business plan. The following section will detail the implementation of a proper information security posture guided by the egrade assessment and other current security principles. The final section will detail the general advances made in securing the website and infrastructure.

Part One: The before snapshot

Current Network Topology

This study focuses on a small business which is a Wireless Internet Service Provider (WISP).The network consists of a number of wireless towers used to provide high-speed Internet access to businesses and homes. All network traffic is aggregated and sent to the Internet on a DS3 fiber connection. Customers are provided with a public IP address that is routable. Some IP addresses are static, most are assigned by DHCP. Each tower typically consists of a router and a number of wireless access points. There are no external firewalls or filters. The e-commerce hardware is located at one of the towers and a firewall is planned for it.

Current hardware configuration

The system contains a 2.4GHz Intel® Celeron® CPU with 256MB physical RAM and 500MB virtual memory, 40GB hard disk drive. Networking hardware includes one Intel Pro1000 NIC and two on-board VIA Rhine III 10/100 NICs.

Current software configuration

The operating system is Linux based distribution (Fedora Core 2) (4) using a default 2.6.8 kernel. This kernel supports loadable modules and is not specifically compiled for the current architecture. Unnecessary services have been turned off and multi user access is enable (eg initlevel = 3, no X-windows) (See Appendix A).

Remote access to the server is accomplished by OpenSSH secure shell code (OpenSSH_3.6.1p2, SSH protocol 2.0, OpenSSL) (5, 6). The default installation configuration is currently being used with the following exceptions; disabled remote root login, protocol 2 enforced, and alternate listening port (see Appendix B). TCP wrappers are employed and configured to deny all hosts (ie ALL:ALL) and accept only three remote IP address for connection to all ports.

In addition to the underlying operating system, the following applications are needed to support the ecommerce website. The MySQL (7) database software was chosen for its price (free) and compatibility with other components of the website. The source code for the latest stable version (4.0.20) was downloaded. Default MySQL rpm packages were removed to prevent confusion (i.e. rpm -e mysq*). MySQL was configured to include SSL support modules and include the following libraries;

```
>>./configure --prefix=/XXX/xxx/MYSQL --with-tcp-port-1000 --with-openssl –with-
excharsets=complex
```

These libraries and other modules in the default configuration will be discussed later as they apply to improving the security posture of the server. The MySQL daemon was started (set to start on system boot), the default database was created, and two users were added and granted all privileges. No security policy is currently enforced on the daemon or at the level of the databases within MySQL.

The Apache web server (8) was chosen to power the website because of its price (free) and the relative ease in making customizations. The latest version (2.0.50) was downloaded, configured and installed in a custom fashion in a non-root directory.

```
>>./configure –prefix /XXX/XXX/APACHE –enable-alias --enable-ssl --enable-so
```

The interconnection between the apache webserver and the MySQL database is based on the PHP scripting engine. The latest stable release (4.3.9) was downloaded and installed in a custom manner.

```
>>./configure –with-apxs2=/cxxx/xxx/xxx/httpd/bin/apxs--enable-mm=shared' \
        '—with-MySQL=/xxx/xxx/MYSQL/--with-cURL=/xxx/xxx/cURL-7.12.2--enable-
        calendar' \
        '—with-openssl--with-pear--enable-sockets--enable-track-vars--enable-versioning--
        with-zlib
```

PHP and the customer environment software requires CURL (10) to be installed and linked with PHP. Again, the latest stable release (7.12) was downloaded, configured and installed.

```
>>   ./configure –with-ssl
```

The customer environment software requires email server capability in order to send messages to customers and database administrators. Sendmail (11) is a popular mail server that is included with most linux distributions. For this site, the original sendmail rpm packages were removed from the machine. In the interest of testing, the latest stable release was downloaded, configured, and installed in the default manner.

```
>>rpm -e sendmail-8.12.11-4.6 sendmail-cf-8.12.11-4.6 mdadm-1.5.0-3 fetchmail-6.2.5-2
mutt-1.4.1-6
>>/bin/sh ./Build   all Configuration: pfx=, os=Linux, rel=2.6.8-1.521, rbase=2, rroot=2.6.8-
1, arch=i686, sfx=, variant=optimized Making in /XX/XXX/sendmail-8.13.1/obj.Linux.2.6.8-
1.521.i686/sendmail
```

Finally the customer environment software was downloaded and installed (Modernbill 4.2.8) (12). The configuration required unfettered access to MySQL to initiate its customer configuration database. As stated previously, the customer environment software is built on PHP, for proprietary reasons a significant portion of the called scripts are encoded. The encoding is based on the zend engine, ioncube PHP accelerator technology (13). The installation procedure was web-based and put the final build in the htdocs directory of the web root directory structure.

Following the initial installation of the above software packages, several other information security software tools were added. These include the host packet filtering program IPTables v1.2.9 (14), intrusion detection software (IDS) Tripwire v2.3.1 (15), and the network detection software (NDS) Snort v2.2.0 (16). Initial packet filtering rules block all ports except 922, 943, and 980. Port 922 allows remote secure shell connections and ports 980 and 943 allow two WWW connections and a secure socket layer WWW interface, respectively. Unmatched packets are dropped, especially icmp requests. Traffic leaving the ecommerce server is not currently filtered.

The Tripwire software package allows for monitoring of all desirable files on the host. Unauthorized changes are recognized and appropriate alerts are made. Currently the software has been installed and a rudimentary configuration has been completed, however, monitoring has been suspended until major software installation and configuration is completed. As a compliment to host base monitoring, the Snort network-monitoring tool has also been installed, but not configured or optimized.

Policies

Important as software and hardware are to an ecommerce server, policy regarding usage, security, and recovery operations transcends the actual machine and is more meaningful in the long run. Currently, a nondisclosure agreement (NDA) is in force between the business owner and the consultant. Other authorizations are of written form by email and verbal communication.

Part Two: Actions Taken

eGrade Scoring

The initial report card and resulting grades for each section are included in the Appendix C. In general the score was low. Several weaknesses stood out, some of which were expected prior to the inspection (eg. firewalls, lack of disaster recovery plans). However, the grade card does not account for or address the size of the business entity, cash flow or planned growth. As such the inspection is very objective. Several items within each category were chosen to focus our enhancement efforts, they are listed below in the categories assigned by the score card.

Physical Location: 60/110
Network: 66/105
Operating System: 52/100
Applications: 0/100
System Administrator: 65/100
Site Policy: 5/90
Credit Card Processor: 0/100
Overall Score: 248/705

Hardware/Location Security Enhancements

This category scored low in the initial assessment primarily because of a lack of backup power and fire alarm/suppression. The building housing the facility is small and fire suppression infrastructure is not economical based on the size of the company. Physical access to the ecommerce server has been restricted to two people. The room and building housing the equipment is secured with standard deadbolt door locks. The small business owner holds all keys and controls access to the facilities. Redundant power supplies, network access, failover server, and external/offsite backup facilities are planned as the company expands its operations. Final score remains at 60/110 which has been deemed acceptable for the current business plan.

Network

The network category graded fairly well. However, the lack of a true hardware firewall and accompanying DMZ brought the score down. A separate hardware-based firewall and proxy server are planned for the future as the operation expands. Currently, the software firewall utility IPTables is being used to packet filter all traffic into the server. As a precaution outbound traffic is now restricted to minimize exploit damage and prevent unauthorized use of the server (ie host hopping). A list of the IPTables rules is shown in Appendix D. A nessus probe was also completed after the initial egrade. A single warning and notice was given. The warning revealed that icmp timestamp information was available and that it should be filtered. The iptables rules were modified to allow safe icmp requests but not allow timestamp information to be available. The full report is given in Appendix E.

Host based IDS includes utilities for system integrity checking (SIV), log file monitoring (LFM), and operating system extenders (OSE)(17). The server lacked an intrusion detection system in the first assessment. This has been rectified in part by installing and configuring the SIV utility Tripwire (15) and using SELinux (18) as an

OSE compiled into the kernel. Log file monitoring will be included at a future time with a utility like Swatch (19). Fine tuning of both configurations is expected to continue once the server is officially deployed. Furthermore, a network base intrusion detection system was installed (Snort)(16) to monitor network traffic coming into the firewall. The internet connection is not redundant and no dialup backdoor exists, although encrypted access for remote administration is allowed from predetermined hosts. The final score is 76/105.

Operating System

This section fared poorly in the initial assessment mostly as a result of no auditing utilities being present. Moreover, a separate development system and a problem notification system were not initially present. However, as stated in the "before snapshot" section, all standard services were disabled and the server was nearly network quiet with the exception of three open ports 922, 981, and 943. Scan results for nmap (20) and amap (21) utilities are presented below.

```
>> nmap -sS -P0 -p 1-64992 xxx.xxx.xxx.xxxt
        Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
        Interesting ports on xxx.xxx.xxx.xxx):
        (The 64992 ports scanned but not shown below are in state: closed)
        Port      State      Service
        922/tcp    open       unknown
        943/tcp    open       unknown
        981/tcp    open       unknown

>>amap xxx.xxx.xxx.xxx 943 922 981
amap v4.7 (www.thc.org) started at 2005-01-15 15:37:58 -
APPLICATION MAP mode
Protocol on xxx.xxx.xxx.xxx:981/tcp matches http
Protocol on xxx.xxx.xxx.xxx:922/tcp matches ssh
Protocol on xxx.xxx.xxx.xxx:922/tcp matches ssh-openssh
Unidentified ports: xxx.xxx.xxx.xxx:943/tcp (total 1).
amap v4.7 finished at 2005-01-15 15:38:0
```

In addition to the standard disabling of services, the following steps had acceptable scores; OS lock down, dedicated use, user access, and base OS. The assessment showed that several easily implemented changes were needed such as: filesystem audit capabilites, a scheduled patch list audit, off site backup, IDS, and disk image recovery. Filesystem auditing and IDS has been implemented using Tripwire and SELinux. The patch list is comprised of the mission critical applications and the underlying operating system kernels and libraries. (See Appendix F).

Offsite backups will be maintained and updated daily. The backups will be incremental for each day and full images of the servers hard drive captured weekly. Both backups will be "burned" to non-rewritable media. At the current time, system monitoring and user/application event auditing is not automated, with the exception of email alerts from IDS application, but will be expanded at a future date. A second identical server configuration is also planned as the company expands. The final score in the OS category is a respectable 87/100.

Applications

The initial grade in this category was 15/100 because the applications had only been installed or compiled in a testbed environment. Only two categories were acceptable in the first assessment, non-default installation locations and privacy notices. However, this category is the most important to successful ecommerce operations, so a very detailed description of each line item is required. Each deficient category will be discussed in terms of the individual applications that are present on the server. The first line item in this category was encrypted network connections. Initially communications with the web interface were not encrypted, although all other remote access was encrypted before the first assessment hence the 5/15 score. In production mode the web interface will support secure, encrypted communications with clients via SSL once appropriate certificates have been purchased. Transactions with the credit card companies will also use SSL as required by industry policy (22).

Storage and handling of sensitive information such as credit card numbers requires prudent planning and implementation. Industry policy requires that credit card information that is stored on ecommerce servers to be encrypted and secure. In the current plan, all customer data including credit card numbers will be stored on the same server. Information in the MySQL database will be encrypted and secured per current standards. The customer environment software automatically encrypts its entries into the database. Backups of the database will be encrypted a second time and cataloged with a unique checksum. Finally, credit card numbers will be removed from the database after fulfillment of the transaction although hard copy backups may still retain information on "open" transactions, hence the additional encryption of backed up databases.

The customer environment software handles all transactions through PHP scripts. Many of these scripts are encoded and therefore not readable and can not be analyzed to see if user input is correctly validated. Initial tests showed that the customer input was appropriately sanitized. Error messages were given when obvious mistakes were made. However, fields that accept large strings remain vulnerable to scripting exploits. A recent search of the internet for exploits of this particular application did not reveal problems. Potential scripting problems with the database backend (MySQL) will be discussed below.

The creation of patch lists and the associated auditing had not been completed before the first grade. This process will be integrated into the operating system patch lists and auditing procedures as they are intrinsically linked (ie the apache binary depends on operating system libraries). In addition to in house tracking of application versions and patches, the customer environment software automatically tracks new releases and the relevance of upgrading.

A majority of the source code for the relevant applications was still visible during the first assessment. All code for Apache, MySQL, cURL, PHP, and Sendmail were in the same directory tree as the binaries. However, most of the PHP scripts were encoded and not human readable. Once the applications were compiled and shown to work together in an insecure setting, binaries and source code were separated as described below. Finally, as this company only sells virtual products in cyberspace, inventory tracking is simplified and is properly implemented in the customer environment software.

To increase security beyond what is captured by the egrade assessment in the OS and the application categories, additional kernel modifications were made and CHROOT jails were created and populated as described below.

Kernel

The stock kernel was reconfigured and recompiled (2.6.9-1.11) (23). Only necessary functions were left in the kernel (ie no support for sound, irda, isdn, wireless, ham radio subsystems, etc …). The kernel was recompiled without module support so dynamic module loading is impossible. However, if an intruder could get access to the kernel, loadable modules would be the least of our worries, nonetheless it is another hurdle that must be surmounted by the intruder.

CHROOT Jails

Another effective tool is the chroot jail configuration of a webserver (24). This technique, if correctly implemented, allows the webserver and associated programes (MySQL, PHP, etc...) to all run in an isolated subset of the main filesystem. This allows for complete control of all functions of the webserver associated programs. In the event that a component of the ecommerce server is exploited only commands and utilities that have been intentionally placed in the chroot jail will be accessible to the intruder. This method is not 100% foolproof; exploits have been published (25) for poorly constructed jails. However, it does add another layer to fulfill defense in depth requirements.

I will briefly outline the process in creating and effectively using the chroot jail for this ecommerce application. Resources exist on the web to guide people through the process (24, 26, 27, and 28). As stated previously the server will be using apache, MySQL, and PHP together with the customer environment software. Ideally all the components need to be in the jail. Very restrictive permissions are set (ie least privileges method) so in the event of a failure in any component of the ecommerce server the intruder only inherits the rights of the application which was running as a non root user (ie MySQL was running as a MySQL-user with restricted permissions). Ideally each application was compiled from clean source code outside of the jail, and then copied into the jail. For the current system, the jail was labeled /chroot/httpd/ and each application had its own directory structure under that (eg /chroot/httpd/home/httpd and /chroot/httpd/home/MySQL).

The discovery phase then begins, each application required access to libraries, files, and devices within the original root directory. This is not possible from the chroot jail; one needs to find out which items are required by using the ldd, lsof, and strace commands. Information gained from these commands is presented below (redundant libraries and devices have been removed).

```
>>ldd MySQLd  httpd libPHP4.0
        librt.so.1 => /lib/tls/librt.so.1
        libdl.so.2 => /lib/libdl.so.2
        libssl.so.4 => /lib/libssl.so.4
        libcrypto.so.4 => /lib/libcrypto.so.4
        libpthread.so.0 => /lib/tls/libpthread.so.0
        libz.so.1 => /usr/lib/libz.so.1
        libcrypt.so.1 => /lib/libcrypt.so.1
        libnsl.so.1 => /lib/libnsl.so.1
        libstdc++.so.5 => /usr/lib/libstdc++.so.5
        libm.so.6 => /lib/tls/libm.so.6
        libgcc_s.so.1 => /lib/libgcc_s.so.1
        libc.so.6 => /lib/tls/libc.so.6
        /lib/ld-linux.so.2 => /lib/ld-linux.so.2
        libgssapi_krb5.so.2 => /usr/lib/libgssapi_krb5.so.2
        libkrb5.so.3 => /usr/lib/libkrb5.so.3
        libcom_err.so.2 => /lib/libcom_err.so.2
        libk5crypto.so.3 => /usr/lib/libk5crypto.so.3
        libresolv.so.2 => /lib/libresolv.so.2
        libaprutil-0.so.0 => /lib/libaprutil-0.so.0
        libgdbm.so.2 => /usr/lib/libgdbm.so.2
        libexpat.so.0 => /usr/lib/libexpat.so.0
        libapr-0.so.0 => /lib/libapr-0.so.0
        libMySQLclient.so.12 => /MYSQL/lib/MySQL/libMySQLclient.so.12
        libcURL.so.3 => /usr/local/lib/libcURL.so.3
```

```
>>lsof |grep msyql
MySQLd     1978    MySQL    0r   CHR       1,3             463809 /dev/null
MySQLd     1978    MySQL    1w   REG        3,2     2027    3009759
/xxxx/xxxx/MYSQL/var/xxxx.xxxx.com.err
MySQLd     1978    MySQL    2w   REG        3,2     2027    3009759
/xxx/xxxx/MYSQL/var/xxxx.xxxx.com.err
MySQLd     1978    MySQL    3u  IPv4      3870       TCP *:MySQL (LISTEN)
MySQLd     1978    MySQL    4u  unix 0x112b5080 3872 /tmp/MySQL.sock
```

```
>>strace /usr/sbin/chroot/ /chroot/httpd /home/httpd/bin/httpd –t /
home/httpd/conf/htppd.conf
open("/usr/local/lib/libcURL.so.3", O_RDONLY) = 4
gettimeofday({1104877007, 257547}, NULL) = 0
open("/etc/localtime", O_RDONLY)       = 4
fstat64(4, {st_mode=S_IFREG|0644, st_size=844, ...}) = 0
mmap2(NULL, 4096, PROT_READ|PROT_WRITE,
read(4, "TZif\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\\5\0"..., 4096) = 844
close(4)                        = 0
munmap(0xf6ffe000, 4096)               = 0
write(2, "[Tue Jan 04 14:16:47 2005] [debu"..., 76[Tue Jan 04 14:16:47 2005] [debug]
mod_so.c(247): loaded module PHP4_module) = 76
```

The ldd utility shows all the libraries the application depends upon. Lsof shows devices such as sockets and files that are locked by the application, and strace gives a complete log of all the calls made by the application to the kernel. Each library, device, and utility needs to be copied from the standard root directory into the jail or recreated in the case of /dev/null and /dev/zero. During the population of the chroot jail, the strace command proved useful in tracking down a misconfigured /dev/null device and incorrect directory permissions. In the first case the major and minor numbers of the device where incorrect and the MySQL user did not have permission to write in the /MSQL/var directory both of which prevented the startup of the MySQL daemon.

Each of the services was tested in the chroot jail and was shown to be running in the jail with the appropriate permissions and as the appropriate users. The sendmail application was set up as a “send only” service. A snapshot of several applications running in the chroot jail is shown below.

```
>>ps aux | grep httpd
apache    3677  0.0  1.4 10576 3772 ?        S    16:42   0:00 /home/httpd/bin/httpd -f
/home/httpd/conf/httpd.conf
>>ps aux | grep MySQL
MySQL     2385  0.0  5.3 115088 13600 ?      S    Jan06   0:00
/home/bbennion/MYSQL/libexec/MySQLd
```

As stated previously, the root directory is actually /chroot/httpd/, so the two examples are indeed running in the appropriate chroot jail.

Application Exploits

Each application in the ecommerce server has its own set of known exploits. For example, cross-site scripting and sql-injection are still possible problems in the current setup. Denial of service attacks on the Apache server, are also possible and could not be addressed in the small business context. Moreover, the network the ecommerce server uses is also vulnerable to possible but unlikely TCP session hijacking and 'man in the middle attacks'. In addition, buffer overflows/underruns are discovered continually in application modules and libraries. To increase the security of the server the chroot jail was constructed as an enclosure around the aforementioned problems. Care was taken to remove all visible source code, run all applications as non-root users, and not introduce compilers and interpreters (python, perl, etc ...) that would aid in a break out. The final score score in Applications category is now a more respectable 80/100, although with the implementation of chroot jails the actual score could be higher.

System Administrator

The initial score for this category was 65/100. The fair grade was directly linked to the part time employment of the system administrator. Although the individual is competent and aware of general developments in Linux world, keeping track of all Apache, MySQL, and PHP developments and exploits is a full time position. Therefore, the initial score remains unchanged.

Site Policies

The first egrade for this category was 5/90. The egrade card was specific in vulnerabilties in three major areas, Applications, Operating Systems, and Networks. Disaster recovery and usage plans are being composed and discussed using the SANS Security Essentials course material (28) as models to address these three items. However, no line items in the egrade checklist cover policies regarding system administration-management interactions relative to the operation of the ecommerce site. Therefore the initial score was low because it did not account for the previously signed nondisclosure agreement and other written communications. In starting the company, liability insurance was purchased and in force prior to the first assessment. As stated in the operating system discussion above, IDS alerts will have been implemented. Currently only two individuals have administrative access to the ecommerce server which simplifies the various user policies. All policies (password, login, etc ...) are to be reviewed yearly or as expansion of the business demands.

Policies now in place allow the administrators to shut down the website for security updates and intrusions at any time with notification of management following as soon as possible. Depending on the reason for the closure of the ecommerce site, restarting requires approval from management and the system administrator. Periodic testing of vulnerabilities, external security audits, and automatic network scans (using tools as nessus (30), nmap (20), amap (21) etc...) is required and notice will be given to management as to when the tests and audits will take place. As part of the disaster recovery plans, customers will be notified in the event of a breach in confidentiality so credit reporting agencies can be notified. Appropriate law enforcement authorities will also be notified if customer data has been shown to be released to the public domain. The final egrade for this category was 60/90 which is respectable.

Credit Card Processor

The first egrade was 0/100 due to the fact that a credit card processor had had not been chosen. The customer environment software allows the administrator to choose a processor from a group of merchants is pre-qualified by the software company. Each merchant therefore meets each of the three criteria in this category, and the final grade for this category is 100/100 as it should be.

Part Three: Conclusions

Did the ecommerce grade card enhance the security posture of the ecommerce website? The initial assessment found a total score of 248/705 which was obviously unacceptable. However, with several compromises on physical hardware and intense focus on operating system and application "hardening" the security posture was improved (558/705) (See Appendix G). The strength of the egrade assessment was in its objectiveness. For example the actual checklist was blind to the size of the actual company or its products. On the other hand, development of ecommerce grade cards that implement selections/options that address company size and potential growth would be desirable. Finally, during the case study nearly every principle that was discussed in the actual Security Essentials course was reviewed and implemented where feasible.

References:

1) McAllister, Andrew. “Inspection Grade Card for Conducting Commerce.” SANS Institute Reading Room 27 Aug. 2003 14 Jan. 2005 <http://www.sans.org/rr/whitepapers/ecommerce/570.PHP>.

2) Bricklin, Dan. “Small Business and Websites” 2004. 14 Jan. 2005 <http://www.bricklin.com/smallbusiness.htm>.

3) Hewlett Packard Support Page 2005 14 Jan. 2005 <http://www.hp.com/>.

4) Fedora Core Home Page 2004. 14 Jan. 2005 <http://fedora.redhat.com>.

5) Openssh Home Page 2004. 14 Jan. 2005 <http://www.openssh.org>.

6) Openssl Home Page 2004. 14 Jan. 2005 <http://www.openssl.org>.

7) MySQL Home Page 2005. 14 Jan. 2005 <http://www.MySQL.com>.

8) Apache Home Page 2005. 14 Jan. 2005 <http:///www.apache.org>.

9) PHP Home Page 2005. 14 Jan. 2005 <http://www.PHP.net>.

10) CURL Home Page 2005. 14 Jan. 2005 <http://culr.haxx.se>.

11) Sendmail Home Page 2005 14 Jan 2005 <http://www.sendmail.org/>.

12) ModernGigabyte, LLC Home Page “Modernbill: Customer Environment and Tracking Software”
14 Jan. 2005 <http://www.modernbill.com/home/index.htm>.

13) Ioncube PHP Accelerator Home Page 2005. 14 Jan. 2005 <http://www.PHP-accelerator.co.uk/>.

14) Netfilter Home Page 2004. “Netfilter/Iptables.” 14 Jan. 2005 <http:///www.netfilter.org/>.

15) Tripwire Inc Home Page 2004. “Tripwire: Filesystem monitoring and auditing.” 14 Jan. 2005. <http://www.tripwire.com/>.

16) Snort Home Page 2004 “The open source network intrusion detection system.” 14 Jan. 2005. <http://www.snort.org>.

17) Prohorenko, Alexander “Open source intrusion detection: No-cost system l ockdown.” Devx.com. 9 Nov 2004. 14 Jan. 2005 <http://www.devx.com/security/Article/22442/0/page/2>.

18) National Security Agency-Central Security Service "SELinux" 14 Jan. 2005 <http://www.nsa.gov/selinux/>.

19) Swatch Download Page 2004. "SWATCH: The Simple WATCHer of Logfiles." 14 Jan. 2005. <http://swatch.sourceforge.net/>.

20) Insecure.org Home Page 2004. "nmap:free open source utility for network exploration or security auditing." 14 Jan. 2005. <http://www.insecure.org/nmap/>

21) The Hackers Choice Home Page 2004. "amap: the open source banner and application scanner." 14 Jan 2005. <http://www.thc.org/releases.PHP>.

22) Mastercard Security Page 2005. 14 Jan. 2005 <http://www.mastercardmerchant.com/preventing_fraud/website_security.html>.

23) Linux Kernel Archives Home Page. 14 Jan. 2005 <http://www.Kernel.org>.

24) Peters, Mike."Chrooting Apache." Linux.com. 27 May 2004. 14 Jan 2005 <http://www.linux.com/article.pl?sid=04/05/24/1450203>.

25) Simon, ES Personal Home Page "Breaking chroot jails" 14 Jan. 2005 <http://www.bpfh.net/simes/computing/chroot-break.html>.

26) Peters, Mike. "Securing MySQL." Linux.com. 19 Aug. 2004. 14 Jan. 2005 <http://security.linux.com/security/04/08/19/1422204.shtml?tid=2&tid=74>.

27) Unknown "Chroot'ing sendmail." 14 Jan. 2005 <http://www.pagasa.net/mail/chroot/chroot.html>.

28) Zdziarski,Jonathan A. "Chrooting daemons and system processes HOW-TO." Linuxexposed.com 5 June 2003 14 Jan. 2005 <http://www.linuxexposed.com/modules.PHP?op=modload&name=News&file=article&sid=506>.

29) Sans Institute. Track 1 -Security Essentials Sans Press Jan. 2004.

30) Nessus Home Page 2004. "The open source vulnerability scanner project." 14 Jan. 2005 <http://www.nessus.org>.

Appendix A

```
# /sbin/chkconfig --list
tux             0:off   1:off   2:off   3:off   4:off   5:off   6:off
kudzu           0:off   1:off   2:off   3:on    4:on    5:on    6:off
sshd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
nfs             0:off   1:off   2:off   3:off   4:off   5:off   6:off
irda            0:off   1:off   2:off   3:off   4:off   5:off   6:off
rpcgssd         0:on    1:off   2:off   3:off   4:off   5:off   6:on
netdump         0:off   1:off   2:off   3:off   4:off   5:off   6:off
irqbalance      0:off   1:off   2:off   3:off   4:off   5:off   6:off
iptables        0:off   1:off   2:on    3:on    4:on    5:on    6:off
gpm             0:off   1:off   2:on    3:on    4:on    5:on    6:off
dc_server       0:off   1:off   2:off   3:off   4:off   5:off   6:off
netfs           0:off   1:off   2:off   3:off   4:off   5:off   6:off
lisa            0:off   1:off   2:off   3:off   4:off   5:off   6:off
netplugd        0:off   1:off   2:off   3:off   4:off   5:off   6:off
cpuspeed        0:off   1:on    2:on    3:on    4:on    5:on    6:off
nfslock         0:off   1:off   2:off   3:off   4:off   5:off   6:off
saslauthd       0:off   1:off   2:off   3:off   4:off   5:off   6:off
snmptrapd       0:off   1:off   2:off   3:off   4:off   5:off   6:off
rhnsd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
cups            0:off   1:off   2:off   3:off   4:off   5:off   6:off
anacron         0:off   1:off   2:on    3:on    4:on    5:on    6:off
acpid           0:off   1:off   2:off   3:off   4:off   5:off   6:off
rpcsvcgssd      0:on    1:off   2:off   3:off   4:off   5:off   6:on
isdn            0:off   1:off   2:off   3:off   4:off   5:off   6:off
autofs          0:off   1:off   2:off   3:on    4:on    5:on    6:off
aep1000         0:off   1:off   2:off   3:off   4:off   5:off   6:off
atd             0:off   1:off   2:off   3:on    4:on    5:on    6:off
winbind         0:off   1:off   2:off   3:off   4:off   5:off   6:off
ntpd            0:off   1:off   2:off   3:off   4:off   5:off   6:off
httpd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
syslog          0:off   1:off   2:on    3:on    4:on    5:on    6:off
dc_client       0:off   1:off   2:off   3:off   4:off   5:off   6:off
yum             0:off   1:off   2:off   3:off   4:off   5:off   6:off
microcode_ctl   0:off   1:off   2:off   3:on    4:on    5:on    6:off
smb             0:off   1:off   2:off   3:off   4:off   5:off   6:off
bcm5820         0:off   1:off   2:off   3:off   4:off   5:off   6:off
random          0:off   1:off   2:on    3:on    4:on    5:on    6:off
rpcidmapd       0:on    1:off   2:off   3:on    4:off   5:on    6:on
vncserver       0:off   1:off   2:off   3:off   4:off   5:off   6:off
readahead       0:off   1:off   2:off   3:off   4:off   5:on    6:off
xfs             0:off   1:off   2:on    3:on    4:on    5:on    6:off
pcmcia          0:off   1:off   2:off   3:off   4:off   5:off   6:off
network         0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

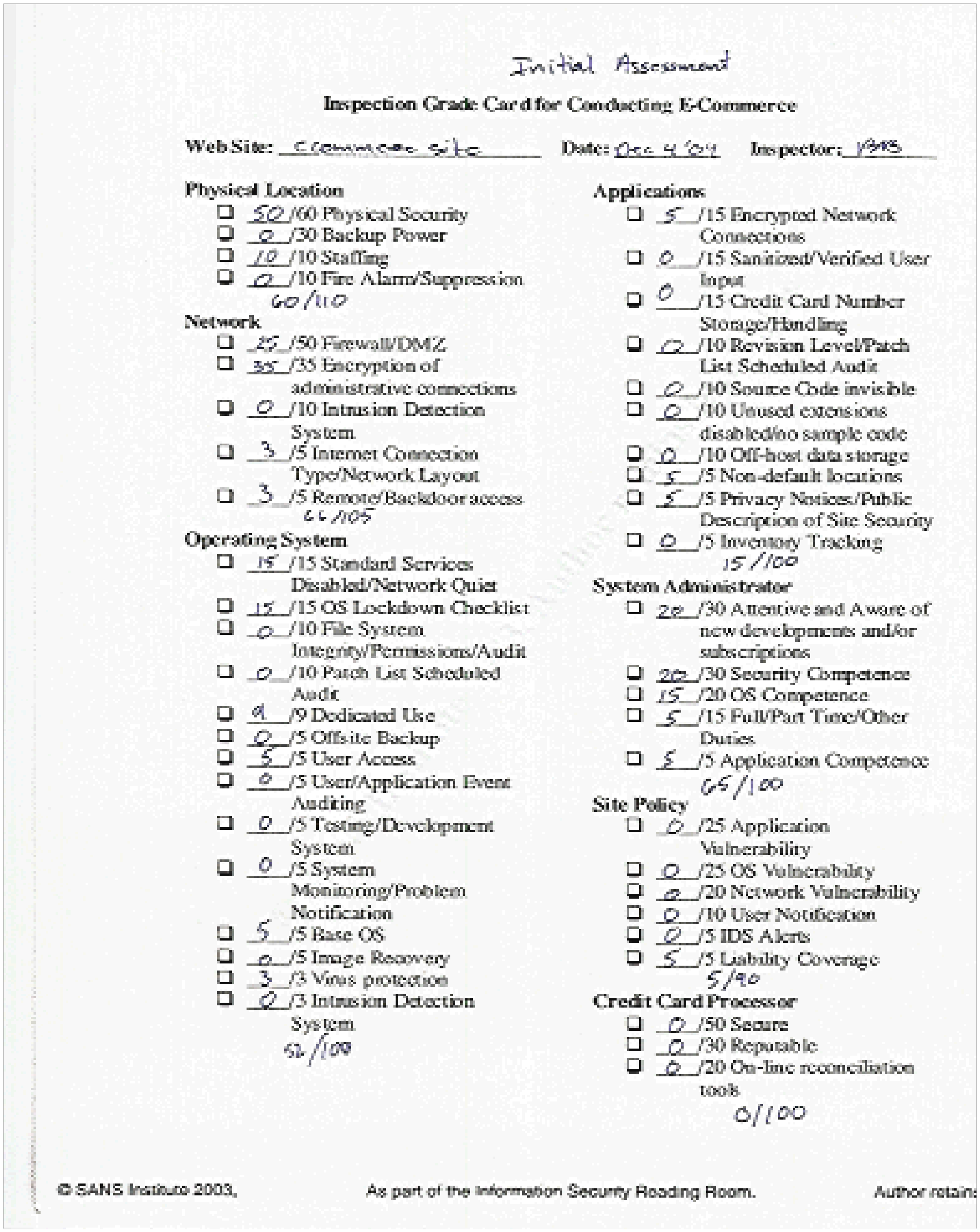
```
squid           0:off   1:off   2:off   3:off   4:off   5:off   6:off
portmap         0:off   1:off   2:off   3:off   4:off   5:off   6:off
xinetd          0:off   1:off   2:off   3:off   4:off   5:off   6:off
sysstat         0:off   1:on    2:on    3:on    4:on    5:on    6:off
apmd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
smartd          0:off   1:off   2:on    3:on    4:on    5:on    6:off
messagebus      0:off   1:off   2:off   3:on    4:on    5:on    6:off
snmpd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
psacct          0:off   1:off   2:off   3:off   4:off   5:off   6:off
rawdevices      0:off   1:off   2:off   3:on    4:on    5:on    6:off
named           0:off   1:off   2:off   3:off   4:off   5:off   6:off
readahead_early 0:off   1:off   2:off   3:off   4:off   5:on    6:off
```

Appendix B

(SSHD initial Configuration irrelevant information deleted)

```
#	$OpenBSD: sshd_config,v 1.59 2002/09/25 11:17:16 markus Exp $
# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.
# This sshd was compiled with PATH=/usr/local/bin:/bin:/usr/bin
# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options change a
# default value.
Protocol 2
ListenAddress XXX.XXX.XXX.XXX:922
SyslogFacility AUTHPRIV
LogLevel INFO
# Authentication:
LoginGraceTime 120
PermitRootLogin no
StrictModes yes
RhostsAuthentication no
# Don't read the user's ~/.rhosts and ~/.shosts files
IgnoreRhosts yes
X11Forwarding yes
PrintLastLog yes
Subsystem	sftp	/usr/libexec/openssh/sftp-server
```

Appendix C
(Initial assessment)



Initial Assessment

Inspection Grade Card for Conducting E-Commerce

Web Site: ecommerce site **Date:** Dec 4 '04 **Inspector:** JSMS

Physical Location
- □ 50 /60 Physical Security
- □ 0 /30 Backup Power
- □ 10 /10 Staffing
- □ 0 /10 Fire Alarm/Suppression

60/110

Network
- □ 25 /50 Firewall/DMZ
- □ 35 /35 Encryption of administrative connections
- □ 0 /10 Intrusion Detection System
- □ 3 /5 Internet Connection Type/Network Layout
- □ 3 /5 Remote/Backdoor access

66/105

Operating System
- □ 15 /15 Standard Services Disabled/Network Quiet
- □ 15 /15 OS Lockdown Checklist
- □ 0 /10 File System Integrity/Permissions/Audit
- □ 0 /10 Patch List Scheduled Audit
- □ 9 /9 Dedicated Use
- □ 0 /5 Offsite Backup
- □ 5 /5 User Access
- □ 0 /5 User/Application Event Auditing
- □ 0 /5 Testing/Development System
- □ 0 /5 System Monitoring/Problem Notification
- □ 5 /5 Base OS
- □ 0 /5 Image Recovery
- □ 3 /3 Virus protection
- □ 0 /3 Intrusion Detection System

52/100

Applications
- □ 5 /15 Encrypted Network Connections
- □ 0 /15 Sanitized/Verified User Input
- □ 0 /15 Credit Card Number Storage/Handling
- □ 0 /10 Revision Level/Patch List Scheduled Audit
- □ 0 /10 Source Code invisible
- □ 0 /10 Unused extensions disabled/no sample code
- □ 0 /10 Off-host data storage
- □ 5 /5 Non-default locations
- □ 5 /5 Privacy Notices/Public Description of Site Security
- □ 0 /5 Inventory Tracking

15/100

System Administrator
- □ 20 /30 Attentive and Aware of new developments and/or subscriptions
- □ 20 /30 Security Competence
- □ 15 /20 OS Competence
- □ 5 /15 Full/Part Time/Other Duties
- □ 5 /5 Application Competence

65/100

Site Policy
- □ 0 /25 Application Vulnerability
- □ 0 /25 OS Vulnerability
- □ 0 /20 Network Vulnerability
- □ 0 /10 User Notification
- □ 0 /5 IDS Alerts
- □ 5 /5 Liability Coverage

5/90

Credit Card Processor
- □ 0 /50 Secure
- □ 0 /30 Reputable
- □ 0 /20 On-line reconciliation tools

0/100

Appendix D

(Iptables configuration)

```
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [3483:445074]
:RH-Firewall-1-INPUT - [0:0]
-A INPUT -j RH-Firewall-1-INPUT
-A FORWARD -j RH-Firewall-1-INPUT
-A OUTPUT -p icmp -m icmp --icmp-type 8 -m state --state NEW -j ACCEPT
-A OUTPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 981 -m state --state NEW -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 922 -m state --state NEW -j ACCEPT
-A OUTPUT -p tcp -m tcp --dport 943 -m state --state NEW -j ACCEPT
-A OUTPUT -j REJECT --reject-with icmp-host-prohibited
-A RH-Firewall-1-INPUT -i lo -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type 0 -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type 3 -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type 4 -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type 11 -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type 12 -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type 14 -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type 16 -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -m icmp --icmp-type 18 -j ACCEPT
-A RH-Firewall-1-INPUT -p icmp -j DROP
-A RH-Firewall-1-INPUT -p ah -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 981 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 943 -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp -m state --state NEW -m tcp --dport 922 -j ACCEPT
-A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

Appendix E

Nessus Scan Report

```
------------------
SUMMARY

 - Number of hosts which were alive during the test : 1
 - Number of security holes found : 0
 - Number of security warnings found : 0
 - Number of security notes found : 1

TESTED HOSTS

DETAILS

+ XXX.XXX.XXX.XXX :
 . List of open ports :
   o general/udp (Security notes found)

 . Information found on port general/udp
    For your information, here is the traceroute to XXX.XXX.XXX.XXX :
    XXX.XXX.XXX.XXX
    XXX.XXX.XXX.XXX
    ------------------------------------------------------
```

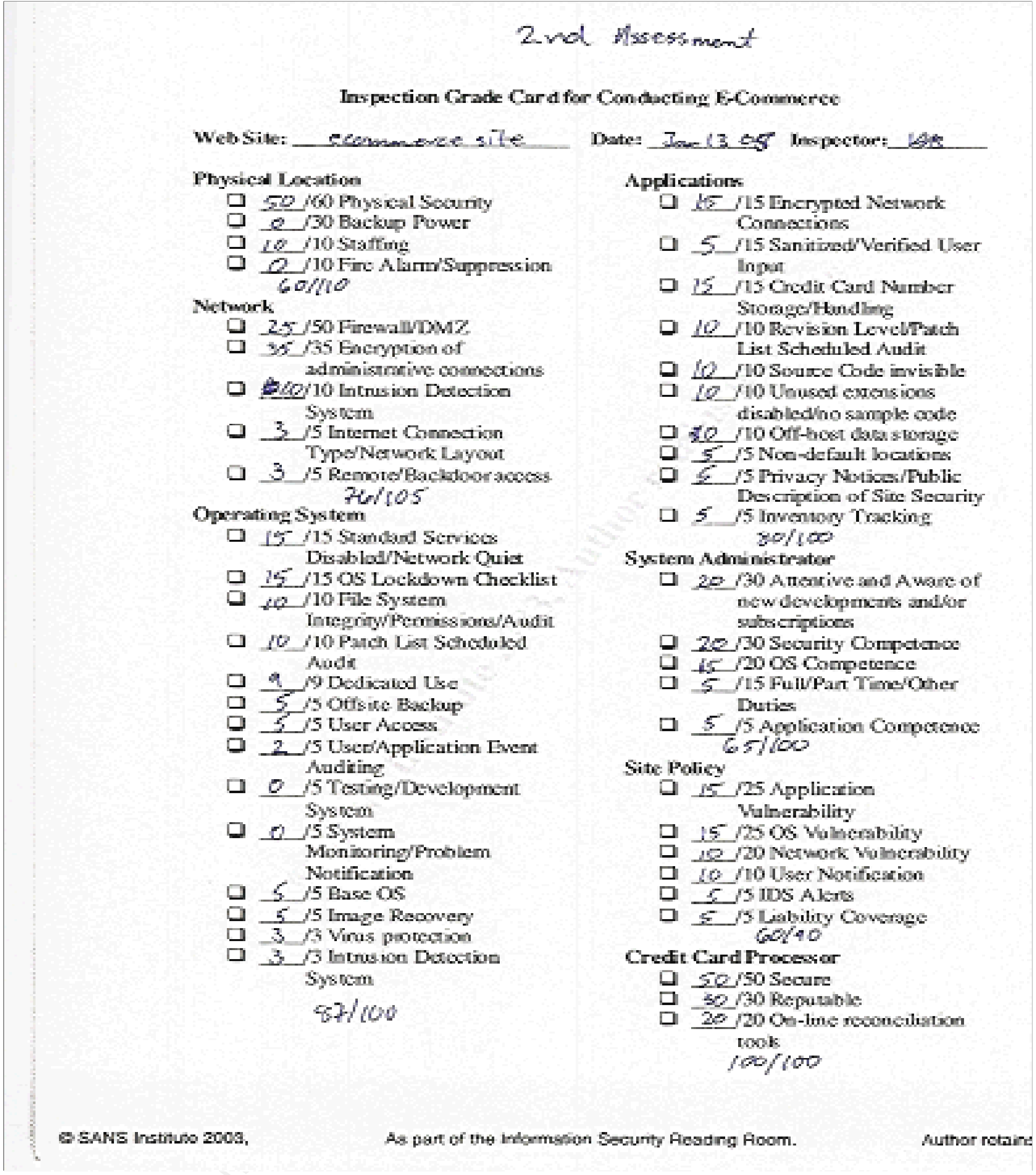
This file was generated by the Nessus Security Scanner

Appendix F

Patchlists for January 2005

Application	Installed Version	Current Version
Apache	2.0.52	2.0.52
MySQL	4.0.20	4.1.9
cURL	7.12.2	7.12.2
gcc	3.3.3	3.4.3
PHP	4.3.9	4.3.10
Sendmail	8.13.1	8.13.3
Tripwire	2.3.2-2	2.3.2-2 Academic
Snort	2.2.0	2.2.0
Kernel	2.6.9-1.11	2.6.10
IPTables	1.2.9	1.2.11
glibc	2.3.3	2.3.3
Modernbill	4.2.8	4.2.8

Appendix G
(Final assessment)



2nd Assessment

Inspection Grade Card for Conducting E-Commerce

Web Site: Ecommerce site Date: Jan 13 05 Inspector: LH

Physical Location
- ❑ 50 /60 Physical Security
- ❑ 0 /30 Backup Power
- ❑ 10 /10 Staffing
- ❑ 0 /10 Fire Alarm/Suppression

60/110

Network
- ❑ 25 /50 Firewall/DMZ
- ❑ 35 /35 Encryption of administrative connections
- ❑ 10 /10 Intrusion Detection System
- ❑ 3 /5 Internet Connection Type/Network Layout
- ❑ 3 /5 Remote/Backdoor access

76/105

Operating System
- ❑ 15 /15 Standard Services Disabled/Network Quiet
- ❑ 15 /15 OS Lockdown Checklist
- ❑ 10 /10 File System Integrity/Permissions/Audit
- ❑ 10 /10 Patch List Scheduled Audit
- ❑ 9 /9 Dedicated Use
- ❑ 5 /5 Offsite Backup
- ❑ 5 /5 User Access
- ❑ 2 /5 User/Application Event Auditing
- ❑ 0 /5 Testing/Development System
- ❑ 0 /5 System Monitoring/Problem Notification
- ❑ 5 /5 Base OS
- ❑ 5 /5 Image Recovery
- ❑ 3 /3 Virus protection
- ❑ 3 /3 Intrusion Detection System

87/100

Applications
- ❑ 15 /15 Encrypted Network Connections
- ❑ 5 /15 Sanitized/Verified User Input
- ❑ 15 /15 Credit Card Number Storage/Handling
- ❑ 10 /10 Revision Level/Patch List Scheduled Audit
- ❑ 10 /10 Source Code invisible
- ❑ 10 /10 Unused extensions disabled/no sample code
- ❑ 10 /10 Off-host data storage
- ❑ 5 /5 Non-default locations
- ❑ 5 /5 Privacy Notices/Public Description of Site Security
- ❑ 5 /5 Inventory Tracking

80/100

System Administrator
- ❑ 20 /30 Attentive and Aware of new developments and/or subscriptions
- ❑ 20 /30 Security Competence
- ❑ 15 /20 OS Competence
- ❑ 5 /15 Full/Part Time/Other Duties
- ❑ 5 /5 Application Competence

65/100

Site Policy
- ❑ 15 /25 Application Vulnerability
- ❑ 15 /25 OS Vulnerability
- ❑ 10 /20 Network Vulnerability
- ❑ 10 /10 User Notification
- ❑ 5 /5 IDS Alerts
- ❑ 5 /5 Liability Coverage

60/90

Credit Card Processor
- ❑ 50 /50 Secure
- ❑ 30 /30 Reputable
- ❑ 20 /20 On-line reconciliation tools

100/100