Global Information Assurance Certification Paper

Inspection Grade Card for Conducting E-Commerce

Andrew McAllister

GIAC Level One Security Essentials

GSEC Practical Requirements (v.1.2f)

Inspection Grade Card for Conducting E-Commerce

Web Site: _____ Date: _____

Inspector:_____

Physical Location

- ❑ ____/60 Physical Security
- ❑ ____/30 Backup Power
- ❑ ____/10 Staffing
- ❑ ____/10 Fire Alarm/Suppression

Network

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

Operating System

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

Applications

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

System Administrator

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

Site Policy

- ❑ ____/25 Application Vulnerability
- ❑ ____/25 OS Vulnerability
- ❑ ____/20 Network Vulnerability
- ❑ ____/10 User Notification
- ❑ ____/5 IDS Alerts

- ❑ ____/5 Liability Coverage

Credit Card Processor

- ❑ ____/50 Secure
- ❑ ____/30 Reputable
- ❑ ____/20 On-line reconciliation tools

Introduction and Instructions for the E-Commerce Grade Card

For e-commerce site owners: Using the "grade card" above, have a competent computer security specialist review each e-commerce site in the organization. Descriptions of each category and item to be graded are provided in the text below. Use the descriptions and sample questions to prepare for the inspection. You will receive a letter grade in each of the seven categories. To pass the inspection you must receive a letter grade of "C" (75%) or higher in each category. Higher scores mean less risk for your e-commerce activity. Ideally, every site should receive an overall score of "A".

In some instances the reviewer will perform a specific network test, physical inspection, or code review; in others, the reviewer may make a subjective judgment for each item. The accuracy of this system is in large part dependent on the skills of the grader, the more skilled he or she is, the more constructive their feedback and the safer your site will be. Finally, it is possible that these inspections may disrupt normal site activity. Provide your grader with a written permission letter specifically authorizing whatever tests he or she deems necessary.

For computer security specialists: For every item on the grade card, score the site using a scale of 0 to the maximum number of points listed. The point values above are recommendations; change them as necessary for your local threat conditions. Use whatever testing tools are necessary to evaluate the site's compliance with each item in the list. Total the points awarded for each category, and assign a letter grade using the following scale: A – 100 to 93, B – 92 to 85, C – 84 – 75, F below 75. Due to the differences in network environments, operating system and application software, it is not possible to provide a list of each item that must be verified. Use your skills as a security specialist and the suggestions provided. Note specific deficiencies so that the site owner can make immediate and meaningful corrections. Failure to meet any of the 12 baseline requirements for e-commerce should result in an automatic failure for the applicable category.

Baseline

Visa U.S.A. has a Cardholder Information Security Program[1] with a list of 12 requirements[2] for e-commerce sites who wish to accept Visa branded credit cards. As of May 2001, these requirements are no longer optional. Failure to meet these baseline requirements may be a violation of the Visa U.S.A Operating Regulations, which could eventually subject the site operator to fines or a withdrawal of card acceptance privileges. The requirements are:

1. Install and maintain a working network firewall to protect data accessible via the Internet.
2. Keep security patches up-to-date.
3. Encrypt stored data.
4. Encrypt data sent across open networks.

5. Use and regularly update anti-virus software.
6. Restrict access to data by business "need-to-know." [sic]
7. Assign a unique ID to each person with computer access to data.
8. Don't use vendor-supplied defaults for system passwords and other security parameters.
9. Track access to data by unique ID.
10. Regularly test security systems and processes.
11. Maintain a policy that addresses information security for employees and contractors.
12. Restrict physical access to cardholder information.

In addition, Visa and MasterCard have mandated that as of January 2001, each credit card transaction that traverses the Internet or other public network must carry an e-commerce indicator (ECI) flag.[3,4,5] This flag is used to mark transactions so that Visa and MasterCard can distinguish between e-commerce and other "card-not-present" transactions. Failure to transmit the flag can result in fines of $5,000 or more per month for Visa and $1000 per month for MasterCard.[1]

Categories and Items to be Graded

Physical Location

E-commerce servers should be located in a secure environment that is not accessible by the general public. Typical data center security with verification and logging of all individuals when they enter and leave is highly desirable. Video surveillance is a plus. For smaller operations, it is possible to have a secure server located in an office environment, but only if it is observable by more than one attentive employee with strict instructions on handling unknown visitors. In an office environment, server cases, keyboards, drives, etc. should be protected by a locking door or cabinet. Servers should **not** be located in or adjacent to public computing facilities (schools, libraries, universities).

Backup power is a must. In the event of a power failure, appropriate personnel should be notified electronically and, if necessary, systems should be configured to shut down safely. Automatic generator power is recommended; battery power is tolerable.

Each server location should have adequate staff or electronic systems to monitor the

[1] The trade press covered the Electronic Commerce Indicator regulations superficially. Details on the schedule of fines and examples of transactions that must carry the ECI flag can be found in documentation provided by a merchant's bank. The author received a document titled "Visa / MasterCard ECI Rules and Regulations Internet Transactions" [sic] from Commerce Bank. There were no copyrights or confidentiality statements on this paper, but it does not appear to be available directly from Visa U.S.A. Readers are advised to contact their own merchant bank for specific documentation of ECI regulations. The paper detailed a schedule of fines for each acquirer (merchant's bank). For Visa, "...acquirers are allowed 3 months to bring each merchant into compliance." In the 4th month a $5,000 fine is assessed for each non-compliant merchant; in the 5th month $10,000; in the 6th month $25,000. MasterCard fines were $1,000 per merchant identified and $1,000 per month, per merchant for each subsequent month. Up to a maximum of $50,000 per quarter.

physical environment for temperature, humidity, presence of standing or dripping water, etc.

Fire alarms, preferably with links a monitoring service or fire station, and if possible suppression systems designed for computing facilities are highly recommended.

Inspection Recommendations:

1) *Inspectors should attempt to gain access to a facility without proper authorization using social engineering methods: pose as a service technician or new employee. Coordinate with facilities managers* ***prior*** *to the attempt.*
2) *Visit the physical facilities and report deficiencies. Review facility test schedules and pay particular attention to backup power tests.*

Physical Location Questions for site Owners:

1) *Describe the physical environment housing your e-commerce and related systems.*
2) *List the number and types of employees with access to e-commerce equipment.*
3) *What precautions are taken to prevent unauthorized physical access to equipment?*

Network

A firewall with a network demilitarized zone is required and part of the "baseline" for conducting e-commerce.[6] The firewall can be software based, and run on the server itself, but this configuration is not optimal and should be avoided. The firewall:

1) Must be stateful and track individual TCP connections rather than allowing all packets destined to a particular port.
2) Must be configured to allow only network traffic directly related to conducting e-commerce from public networks. In other words, only traffic destined for port 80 or 443 on the web server should be allowed to reach the demilitarized network.
3) Should restrict outbound traffic so that only packets belonging to established connections are allowed out to the public network. For example, it should **not** be possible to telnet from the e-commerce machine to a machine on the public network.
4) Must log all unauthorized traffic to logs on the firewall and should also log to remote files if possible.
5) Must be kept patched and up to date with vendor supplied security patches.
6) Must not allow administrative access to the firewall or DMZ from the public network.

Administrative connections to the network firewall and the e-commerce server must be encrypted at all times and are not allowed from the public network. Remote

administration should use a VPN solution to tunnel administrative traffic across the public Internet or have a direct dial-in connection to the local "trusted" network. From there, administrators should use tools such as SSH or Kerberized applications to connect to hosts. It is **unacceptable** to connect to an e-commerce server with clear text telnet, FTP, RSH, NFS, VNC, etc. regardless of where the connection originates. File access via NFS, SAMBA or windows file sharing is **not** safe. Win2K with file sharing via IPSEC is acceptable.

A network intrusion detection system is a plus and one that modifies the behavior of the network firewall is even more useful. An IDS inside the demilitarized network should monitor odd traffic particularly from or to the public network via the firewall. The firewall itself should log and eliminate much of the traffic that would generate false positives, so the IDS system should be set, at least initially, at a very sensitive level. The ability for an IDS system to check for application layer attacks is particularly important.

The type and speed of connection to the public networks will vary depending on the amount of e-commerce activity. Bigger sites will require redundant paths, higher capacity links, faster firewalls, etc. The network should be built with reliable components and sufficient service contracts or spare parts available so as not to distract network administrators from potential security problems.

Remote administration and backdoor access to systems are discouraged. Direct dial-up connections into servers are inherently dangerous and should be avoided completely. A more appropriate administrative back door would be a VPN solution into the non-public side of the demilitarized zone network, from there, encrypted administrative connections can be made with some safety.

Inspection Recommendations:

1) *Use a network-port scanning tool like nmap[7,8] to probe the site's DMZ from both the public and private networks. Check all open ports and verify that administrative connections are available only from specific locations on the internal network.*
2) *Verify network encryption by installing a packet-sniffer, e.g. tcpdump, and analyzing logs for plain text.[9]*

Network Operations Questions for site Owners:

1) *Provide logical and physical diagrams of network equipment, firewalls, and servers.*
2) *List the network traffic protocol types, IP addresses, and ports allowed into the e-commerce network from the Internet. Repeat for the organization's internal network.*
3) *List all encryption techniques and software packages used.*
4) *Describe how system administrators access the e-commerce systems from outside the organization's internal network.*

5) *List two people who can remove the e-commerce site from the network (Disable network ports etc.) in an emergency situation.*

Operating System

The base operating system of any e-commerce solution can greatly affect the overall security environment. Many operating systems can be made reasonably secure, but doing so may take significantly more effort on some platforms.

Microsoft Windows 3.1, 95, 98, and ME are **not** acceptable e-commerce platforms, period. They cannot be made secure.

Microsoft Windows NT, Windows 2000, and XP are barely tolerable operating systems for e-commerce, and require a significant amount of effort to secure. The installation of vulnerable components in the Windows subsystems generally cannot be controlled easily and leaves many undiscovered holes for crackers to exploit. There is no reason to have a web browser and integrated e-mail client an e-commerce server, yet it is installed by default on all recent versions of Windows. In addition, due to their market position and a general dislike of Microsoft by the hacking/cracking community, Windows NT/2K/XP systems are significant targets for attack.[10] Patches for Windows servers often require the machine to be rebooted, thus interrupting services. This serves as a significant disincentive to timely patch application and makes NT/2K systems even more vulnerable. Additionally, many administrative functions can only be accomplished via a graphical interface, that is, by remote control and/or logging in on the system console; or by sending data over insecure NetBIOS connections. Windows 2000 and presumably XP offer improvements in this area, but these improvements are often disabled to maintain backward compatibility with older software. It is often assumed that Windows systems are easier to administer. In fact, NT/2K systems can be more difficult to administer in a secure environment. Windows NT/2K/XP should be used only when a required e-commerce application is not available on a Unix platform.

Unix operating systems such as Linux, Solaris, HP/UX, and AIX are acceptable for e-commerce. Each OS still requires some effort to make it secure, but generally the techniques to do so are well documented. Source code is often available for Unix software which helps expose flaws, but simultaneously speeds up the subsequent fixes (clearly a two edged sword). Unix system administrators must be diligent in keeping software packages up to date and vendor supplied patches installed. Command line utilities and text configuration files appear to make Unix more difficult to administer. But they are also extremely convenient for remote, low bandwidth, administration and make Unix systems easier to “lock down”. Always use the latest version of a Unix operating system for e-commerce. Brand new releases are sometimes buggy, but are also fixed very quickly.

Every operating system should only run those network services required for an e-commerce application. Telnet, FTP, NFS, SNMP, SMTP, RPC, NNTP, finger, etc. are **not**

essential to e-commerce and should be disabled and preferably removed or not installed at all. On most Unix systems running as web servers, it is possible to completely disable all network services except HTTP and HTTPS. Disabling services **must** be part of the installation process. To every extent possible **all** services should also be disabled on Windows systems. In all cases the DMZ firewall should also block the ports on which these services run just in case the services are enabled as part of a patch or upgrade.

Most operating systems have a lockdown or security checklist that may be followed to make the system more secure. Microsoft provides one for NT[11] and IIS4[12], as well as tools for Win2K and IIS5[13], but a more extensive lockdown checklist is also available from SANS.org as part of their coursework. SANS also has a checklist for Linux variants. Sun documents can be found on several web sites[14,15] and the book "Solaris Security" by Peter H. Gregory[16] is also extremely useful. Following the steps in one or more security checklists is required **before** an e-commerce server is connected to any network.

Plugging security holes is useful, but without an extensive file system audit tool, a hacker could sneak in through a previously unknown hole and replace critical system files thus opening more security holes. The use of a file-system monitoring tool like Tripwire is required for an e-commerce server.[17] Versions of Tripwire are available on Windows NT/2K, Linux, and Solaris, AIX, and HP/UX. Tripwire 1.3.x academic version is unacceptable because its integrity databases are not encrypted. Instead, use Tripwire 2.2.x or higher. Tripwire integrity checks should be run often and system administrators **must** be given time to analyze the reports and update the integrity databases.

Operating system patches **must** be applied as they are released; this is part of the Visa Cardholder Information Security Program.[18] Security patches should be applied immediately upon release unless they are reported to be buggy, other patches should be applied at least weekly. An example of the need for timely patch application can be found in the "Code Red" worm released in the middle of July 2001. In just a few days, this network worm made its way into over 225,000 Windows IIS web servers all over the Internet through a bug in a native component of Windows 2000 (an add-on for NT).[19] Microsoft had previously released a fix for this security flaw on June 18, 2001, almost one month earlier. A patch audit utility or script should be downloaded or created to verify which patches are available and which are installed on each e-commerce server. Patch audits should be scheduled as often as vendors update their web sites with patch information, usually daily or weekly.

E-commerce servers **must** be dedicated to one purpose, E-commerce. They cannot also be used as database servers, file servers, print servers, network storage, routers, etc.[20] Running more than one activity on a server almost always requires a compromise of security settings. Just as a cash register in a normal store should not also be used to play games or surf the web, an e-commerce server should only be used for accepting orders and keeping customer information safe. This is a cost of doing business on the web.

Other general security precautions should be taken with e-commerce servers. These include: 1) limiting the number of users with access to the server, except for the purpose of e-commerce. 2) Making timely and secure off-site backups. 3) Auditing user actions and system events combined with an automatic notification system for administrators. 4) Using virus protection (particularly important on Windows platforms) and intrusion detection tools.

Private test and development systems that are identical to production e-commerce sites can dramatically improve a site's reliability and security. Often, if the systems are truly identical, it is possible to apply OS or application patches to one system, test them and then bring the patched system on-line to replace the production system. The production system can then be patched or held in reserve if the patch is defective. If the e-commerce server acts only as a web transaction server, it is NOT necessary to spend great sums on redundant hardware and huge storage sub-systems. Often times, two inexpensive servers can be had for the price of one complex and highly redundant system. A clever system administrator can then use conventional techniques to bring one machine online in the event of a system compromise (fixing the hole on the backup machine first of course).

Inspection Recommendations:

1) *Verify that the appropriate checklists have been used. Request a copy of each machine's completed checklist and randomly verify the items on the list.*
2) *Search the vulnerability database at SecurityFocus.com[21] and confirm that all known vulnerabilities have been fixed (assign task to local system admin).*
3) *Scan the Operating system from inside and outside the network DMZ with a security scanner like Nessus.[22]*

Operating System Questions for site Owners:

1) *List all operating systems involved in e-commerce operations. Describe each machine in terms of its OS, and its role in operations.*
2) *List all patches installed on each machine. List all current patches available from the vendor for each OS. Justify discrepancies.*
3) *List all virus scanners, audit, and intrusion detection tools in use (including versions). Describe how and when they are used, and what is done with the results.*
4) *List all users with access to the e-commerce servers. Describe their level of access and their role in operations.*

Applications

Some of the fastest growing items of security concern in recent times are e-commerce applications themselves. As firewalls, operating systems, and system administrators get more diligent, crackers or hackers are instead turning to parts of the computer network that by their nature **must** remain open. Namely, web servers and their applications. By

far, the most commonly exploited applications run on Windows NT/2000 and IIS (Microsoft's Web server). In fact, the IIS server itself contributes a significant level of risk to any e-commerce application.[23]

First and foremost, VISA, MasterCard, and other credit card company's terms of use, require all network traffic that carries credit card transaction information across a public network to be encrypted with industry standard techniques.[24] Typically this means: 1) Credit card information must be transmitted between the shopper's browser and e-commerce server using HTTPS, typically encrypted with SSL version 2 or higher with 128-bits of encryption. 2) Assuming that the credit card information leaves the e-commerce server for further processing (even just to print it out) that network traffic **must** also be encrypted. It is absolutely **unacceptable** to send credit card information through e-mail for any reason. Furthermore, credit card companies have expanded their regulations to require that credit card numbers stored on disk (in a database etc.) must also be encrypted.[25]

Fundamentally, application security requires that every piece of data sent to a web server must be handled in a way that does not produce unexpected results. In reality, this is extremely difficult.

In a typical e-commerce transaction, a web server sends an HTML page to a browser that draws a "checkout" screen, requesting credit card and shipping information for the user's order. The user enters the data and hits a submit button or icon and the data is packaged neatly and sent back to the server for processing. The process seems simple, but can go awry particularly when functionality is added to the web client through the use of client side JavaScript. For example, assume for a moment that a shopping cart checkout screen contains a listing of the items that make up the order and spaces for the user to change the quantity of those items. A poorly coded application may also hide on that same screen the **price** of those items so that when the user changes the quantity, the total order price is instantly updated. Assume also, that all of this information is sent back to the e-commerce site to place the final order, but never checked against the stored price in the server database. Neat, but trivial for a hacker to simply save the web page out to disk, change the prices, and then reload and submit the page back to the e-commerce site. Now, a not-so-clever hacker may be able to order $1000 worth of merchandise for $10. In this example, the server assumed that the pricing information was correct while the hacker capitalized on that assumption by manipulating data sent to the client before it was returned to the server. A number of commercial shopping cart applications use this dubious technique or similar ones to handle quantity and pricing information.[26]

Another type of application error occurs when the information sent to the server is completely unexpected either in type or size. It is possible to construct a URL that when sent to the appropriate web server can install and run a program that then goes out and attempts to send the same URL to another web server. An entire program can be stored in a single URL. This is exactly how the "Code Red" worm mentioned above made its way

across the Internet.[27] Depending on how the data is handled this program inside a URL can be used to open further security holes, or reveal sensitive information stored on the server. All of this is done right through a firewall on a standard web connection. Web applications **must** be written to handle a variety of data thrown at them.

If local programmers write an e-commerce application, there is an opportunity to handle this bad data correctly[28], but programmers who are not conditioned to use secure programming techniques may make the situation worse (much worse). Unfortunately, commercial applications and shopping carts may not be much better than those written internally. It is therefore critical that commercial applications be purchased from extremely reputable vendors with on-going service and maintenance contracts, and that e-commerce site owners keep their applications up to date with regular vendor-supplied patches. Just like the operating system, patch and version audits must be conducted on a regular basis.

In the event of an error, secure servers must also not reveal any information about the application itself. Typically, web servers and shopping carts have "debug" messages that print when there is a data or server error. These screens may reveal application source code, the directory in which the source code is stored, etc. This information may then allow a hacker to look for database user ID's or passwords embedded in the source code or may reveal a program elsewhere on the system that the hacker can more easily exploit.

Web server software html pages and script files should be installed in non-default locations, so that automated attacks against known exploits aren't as useful. For example, the "scripts" directory in IIS is a gateway to a number of attacks. Changing the directory name, or deleting it and all other delivered sample programs will reduce the risk of (but not eliminate) many application security holes. At the same time, web servers will often process files with certain extensions as programs instead of static pages (.asp, .php, .shtml, etc.), removing this default functionality is also critical.

E-commerce web servers should avoid using their own disks as database storage for the web application. Static pages, and scripts should be stored on the web server, but customer information, inventory, etc. should be stored in a database server **on a separate machine** that is **not** accessible via the internet. This separate database server should also block everything but database traffic to and from the web server so that if the web server is compromised it is not then a trivial task to compromise the database server. Good programmers will use stored procedures in the database to access information instead of allowing the "web" user id full SQL access to all the underlying database tables.

Web sites should also place conspicuous notices on their pages that describe many of the precautions used to protect customer information.[29] This serves two purposes: 1) It lets customers know that the site owners are taking all reasonable steps to ensure the safety of their information and orders, and 2) it lets hackers know how thorough the site's defenses are. Sites with good defense strategies should **not** be afraid of revealing their most

obvious security measures. Clearly one would not post a highly detailed accounting of site security, but a web page saying that a site does not store credit card numbers might convince a casual hacker to look elsewhere for easier prey.

E-commerce sites that sell items that are **not** available for immediate shipping, e.g. items that are out of stock, must not bill a credit card until those items have shipped (this is called fulfillment). In many situations this means that a credit card number must be stored and charged at a later date. New credit card regulations now require all stored credit card numbers to be encrypted.[30] The procedures and software necessary to encrypt and decrypt the credit card numbers in a hacker resistant way can often be more difficult to implement than a system that only allows people to buy items that are in stock. Sites are strongly discouraged from taking orders that cannot be immediately fulfilled.

Inspection Recommendations:

1) *Inspect application generated HTML pages for signs that critical information is sent to the client, and then not verified upon its return. Look for hidden fields in particular. Verify what each hidden field stores and returns.*
2) *Create HTML POST and GET requests that have exceptionally large data fields or contain unexpected data elements.*
3) *Search the SecurityFocus.com vulnerabilities database for known commercial application exploits.*[31]
4) *Verify encryption techniques if credit card data is stored locally.*

Application Questions for site Owners:

1) *Describe how data sent to and from the user is validated when returned to the checkout software.*
2) *What software is required for a user to conduct an e-commerce transaction? Is the application browser-specific?*
3) *List any commercial applications (with version numbers) in use and provide a list of vendor patches available. Provide a list of installed vendor patches. Justify discrepancies.*
4) *Where is user provided data stored? What data is kept for each transaction? Is it encrypted? How?*
5) *Describe how the site software processes an e-commerce transaction. Be specific. List detailed steps including any areas where data is verified against stored values or sent off-site for processing.*
6) *List all users with application level access to e-commerce operations. List their duties and responsibilities.*

System Administrator

A system administrator is like a retail store manager. He or she can make or break the business. A good administrator knows a great deal about all aspects of an e-commerce

system including networking, operating systems, e-commerce applications and most importantly computer security. He or she should report directly to the company security officer, CIO or **someone other than** the director of sales. The system administrator's job is first and foremost to protect company and customer data. Only after that, can he or she keep an e-commerce site running.

System administrators must be hyper-aware of new developments in computer security and intrusion. They must in many cases "hang out" (silently) with the hackers and subscribe/view many of the numerous vulnerability web sites and listserv's.

It is helpful for an e-commerce system administrator to have formal training in computer security. A SANS certification or security coursework from system vendors is highly desirable. It is especially important that the system administrator be familiar with the e-commerce server's operating system and payment acceptance applications.

Also critical to good system administration is time specifically allocated for review of system logs, application of fixes, and research into on-going developments. This time should be scheduled as part of a normal workweek, with definable goals and deliverables. Good system administrators will no-doubt resist a structured format of this nature, but formal assigned duties and times to perform them can help keep the administrator out of frivolous meetings and keep them from being assigned to distracting projects. However, a delicate balance must be maintained, as good administrators will quickly bore of mundane tasks. Adding "good guy" hacking as part of an administrator's assigned duties can keep the environment compelling.

Inspection Recommendations:

1) *Speak directly to the System administrator. Ask questions about the most recent vulnerabilities on the e-commerce platforms run on each site.*

System Administrator Questions for site Owners:

1) *List all people authorized to act as system administrators on the e-commerce servers.*
2) *List any related course work or certifications held by the system administrators.*
3) *List (in general terms) the day-to-day job responsibilities of each system administrator.*
4) *What tools do the system administrators use to monitor critical systems?*
5) *How can each system administrator be reached 24 hours a day? What call schedule is maintained and who are the primary or backup administrators?*
6) *If available, provide a resume for the primary and backup system administrators.*

Site Policy

Every e-commerce site should be governed by a series of simple written policies that are approved in writing by management. These policies should be kept general in nature and simple for anyone to understand (particularly sales management). Each should stress the importance of customer service in the form of security and privacy over the need to drive sales. Sites that put sales above all else will be short lived and make the parent organization notorious in their industry.

Examples of simplified, but good, security policies include:

1) System administrators have the authority to shut down an e-commerce site at any time to apply security or other critical patches.
2) Administrators should shut down an e-commerce operation when an intrusion is suspected and **must** shut down an e-commerce site when it is confirmed. Any intrusion no, matter how trivial it may appear, is cause for an immediate site shutdown.
3) The resumption or continuance of site operations is at the sole discretion of the security officer and system administrator. Approval from both is required.
4) E-mail or other specific threats, claims of intrusion, or the self-discovery of one or more site vulnerabilities must be taken seriously and also warrants a site shutdown.
5) The safety of customer data takes precedence over continuing site operations. Users whose data may have been released are entitled to be notified immediately so that they may protect themselves from further harm. When a specific set of users cannot be determined, all users must be notified.
6) Specific employees are authorized to test the e-commerce site for vulnerabilities, most importantly vulnerabilities that result in loss of control of the e-commerce site or it's data. Publicized vulnerabilities should be immediately tested against all elements of an e-commerce operation and resolutions to those vulnerabilities developed and implemented quickly. Site shutdown may be necessary prior to the development of a resolution or to apply patches and fixes.
7) External security audits will be conducted on a routine basis.
8) Automated vulnerability scans will be run against an e-commerce infrastructure on a routine basis.

Since e-commerce operations involve a certain degree of risk to the organization's reputation and assets, appropriate measures must be taken to limit or mitigate the organization's financial and legal liability.

Inspection Recommendations:

1) *Review copies of all policies regarding e-commerce. Make sure they emphasize security and the protection of customer information over all other factors. Look for clarity and brevity.*
2) *Look for a clear designation of authority in all site policies.*

Site Policy Questions for site Owners:

1) *List all people with the authority to shut down an e-commerce site.*
2) *List those who may authorize its return to service.*
3) *Provide written copies of all local policies regarding e-commerce activity.*

Credit Card Processor

Internet credit card transaction processors come in all shapes and sizes. Many Internet companies will act as resellers of other transaction processors. In general only the most reputable and stable transaction processors should be used. Good processors will place a heavy emphasis on security as well as ease of use, and will have a strong reputation in the trade press.

The best processors will have manual methods of adding, changing, refunding and reconciling, all the transactions sent to them. Without these transaction management tools, e-commerce sites will almost certainly need to store customer credit card numbers on-site. Storing credit card information on-site adds a significant system administration and programming burden, and should be avoided unless absolutely necessary.

Inspection Recommendations:

1) *Download the credit card processor's software toolkit and documentation. Verify that all data transmissions are encrypted and that data is not stored in local temporary files.*
2) *Review the processor's privacy policy.*
3) *Review the processor's on-line tools for reporting and transaction management.*

Credit Card Processor Questions for site Owners:

1) *Who is your credit card transaction processor?*
2) *Do they support the required e-commerce indicator flag?*
3) *What tools are provided for managing transactional data besides those used to send and receive instant authorization?*

List of References:

[1] Visa U.S.A. "Cardholder Information Security Program." Version 5.5. 2000. URL: http://www.visabrc.com/documents/cisp55.pdf (26 Aug 2001).

[2] Ibid: 7

[3] Wahmpreneur. "ECI Coding Comes Back To Haunt Web Merchants." http://www.wahmpreneur.com/Archives/Jan01/decline.html (26 Aug 2001).

[4] Visa U.S.A. "Visa Targets Consumer Confidence with New Global Secure E-Commerce Initiatives." October 10, 2000.URL: http://www-s2.visa.com/av/news/press_release.ghtml?pr_form_edit=363&edit_file= (26 Aug 2001).

[5] Bailey, Larry; Kendrick, Jerry. "On-Line / Electronic Merchants Must Comply w/ ECI." URL: http://www.internetwest.net/information/cc2.htm (26 Aug 2001)

[6] Visa U.S.A. :9-10

[7] Fyodor. "The Art of Port Scanning." 6 Sept 1997. URL: http://www.fm.cornell.edu/computer/security/nmap-howto-portscan.html (26 Aug 2001).

[8] Fyodor, Nmap home page. 21 Jun 2001. URL: http://www.insecure.org/nmap/ (26 Aug 2001).

[9] TCPDUMP home page. 16 Jul 2001. URL: http://www.tcpdump.org/ (26 Aug 2001).

[10] Green, Thomas C. "Hacking IIS -- how sweet it is." The Register. URL: http://www.theregister.co.uk/content/4/20960.html (26 Aug 2001).

[11] Microsoft. "Windows NT C2 Configuration Checklist." 4 Apr 2000. URL: http://www.microsoft.com/technet/itsolutions/security/tools/c2config.asp (26 Aug 2001).

[12] Microsoft. "IIS 4.0 Security Checklist." 24 Jul 2001. URL: http://www.microsoft.com/technet/itsolutions/security/tools/iischk.asp (26 Aug 2001).

[13] Howard, Michael, Microsoft. "Secure Internet Information Services 5 Checklist." 29 Jun 2000. URL: http://www.microsoft.com/technet/itsolutions/security/tools/iis5chk.asp (26 Aug 2001).

[14] CERT, Carnegie Mellon University. "Installing and securing Solaris 2.6 servers." 14 Jun 2000. URL: http://www.cert.org/security-improvement/implementations/i027.02.html (26 Aug 2001).

[15] Noordergraaf, Alex and Watson, Keith. "Solaris Operating Environment Security." Jan 2000. URL: http://www.sun.com/software/solutions/blueprints/0100/security.pdf (26 Aug 2001).

[16] Gregory, Peter H. "Solaris Security." Upper Saddle River, NJ. Prentice-Hall, Inc. 2000

[17] Beale, Jay, "Tripwire – The Only Way to Really Know." 11 Jul 2000. URL: http://www.securityportal.com/topnews/tripwire20000711.html (26 Aug 2001).

[18] Visa U.S.A. "Cardholder Information Security Program."

[19] Lemon, Summer, "'Code Red' worm exploits Windows NT flaw." 20 July 2001. URL: http://www.infoworld.com/articles/hn/xml/01/07/20/010720hnworm.xml (26 Aug 2001).

[20] Visa U.S.A. "Cardholder Information Security Program."

[21] Security Focus Vulnerabilities database. URL: http://www.securityfocus.com/vdb/ (26 Aug 2001).

[22] Nessus homepage, http://www.nessus.org/ (26 Aug 2001).

[23] Green

[24] Visa U.S.A. "Cardholder Information Security Program."

[25] ibid

[26] X-Force. "Form Tampering Vulnerabilities in Several Web Shopping Cart Applications." 1 Feb 2000. URL: http://xforce.iss.net/alerts/advise42.php (26 Aug 2000).

[27] The SANS Institute. "Code Red (II)." Version 0.7. 7 Aug 2001. URL: http://www.incidents.org/react/code_redII.php (26 Aug 2000).

[28] Wheeler, David A. "Chapter 5. Avoid Buffer Overflow." Secure Programming for Linux and Unix HOWTO. Version 2.70. 1 Jan 2001. URL: http://www.linuxdoc.org/HOWTO/Secure-Programs-HOWTO/buffer-overflow.html (26 Aug 2001).

[29] Visa U.S.A. "Web Merchant News." URL: http://www-s2.visa.com/fb/wmn/main.html (26 Aug 2001).

[30] Visa U.S.A. "Cardholder Information Security Program."

[31] Security Focus Vulnerabilities database