

# **White Paper: Managing Vulnerabilities For PCI Compliance**

**By:**

**Christopher S. Harper  
Vice President of Technical Services  
Secure Enterprise Computing, Inc**

**March 1, 2009**



## **NOTE CONCERNING INTELLECTUAL PROPERTY AND SOLUTIONS OF SECURE ENTERPRISE COMPUTING, INC.**

Copyright © 2009 By Secure Enterprise Computing, Inc. All Rights Reserved.

Reproduction and distribution of this publication in any form without prior written permission is forbidden. The information contained herein has been obtained from sources believed to be reliable. Secure Enterprise Computing, Inc. disclaims all warranties as to the accuracy, completeness or adequacy of such information. Although Secure Enterprise Computing's Inc. research may discuss legal and/or compliance issues related to the information technology business, Secure Enterprise Computing, Inc. does not provide legal advice or services and its research should not be construed or used as such. Secure Enterprise Computing, Inc. shall have no liability for errors, omissions or inadequacies in the information contained herein or for interpretations thereof. The opinions expressed herein are subject to change without notice.

## White Paper: Managing Vulnerabilities For PCI Compliance

Christopher S. Harper, Vice President of Technical Services, Secure Enterprise Computing, Inc.

Settling on a PCI vulnerability management strategy is sometimes a difficult task, especially for smaller merchants or those who are in the early stages of achieving compliance. In this short paper, we will outline the requirements in the PCI DSS that are related to identifying, mitigating, and reporting on security vulnerabilities, and recommend a strategy for maintaining compliance in this regard. Before outlining the strategy, it is important to highlight a few things that are essential to managing vulnerabilities (and compliance in general) in a PCI world.

First, be sure to take a risk-based approach to meeting the standard. This requires an understanding, by IT and security managers working with business stakeholders, of the biggest threats to the business and then setting priorities for security and compliance efforts. An added benefit to this approach is that you help to ensure that the security budget is spent wisely – a smart move when dealing with the reality of our current economy. Traditional security spending has taken a more tactical approach, with the mindset of “more security products will certainly bring better security”. This is not the best approach when working towards compliance with a complex standard such as the PCI DSS.

Second, understand and approach the standard holistically, instead of addressing each requirement in a vacuum. If you step back from the standard, it is important to remember that the reason it exists in the first place is to protect cardholder data. Too often we see organizations getting wound up in a specific sub-requirement and what the interpretation should be in their case. Each of the requirements must be addressed and ultimately met; however, they should all be seen as a combined effort to protect data. In this paper we are focused on vulnerability management, and there are many items in the standard that contribute and work together to achieve the goal of reducing the threat of vulnerabilities.

Finally, make sure that your compliance efforts are not static. PCI compliance should be achieved and then maintained. Vulnerability management is no exception. There will be a constant flow of threats to your organization. Managing this environment will require constant vigilance. Whether it is patch management, vulnerability assessment, host hardening techniques, standardization of configurations, training, or policy--all will require a continuous effort to be successful.

### Vulnerability Management and the PCI DSS

Now, let's examine some of the PCI requirements related to managing vulnerabilities:

Requirement 2.2 states that an organization should “Develop configuration standards for all system components. Assure that these standards address all known security vulnerabilities and are consistent with industry-accepted system hardening standards.”

One of the first defenses against security vulnerabilities involves how hosts or devices are configured. This “system hardening” involves configuring the device in such a way that it is specific to task and contains no superfluous components or settings that may be vulnerable now or in the future. This is a great way to eliminate problems before they arise.

Requirement 6.1 reads “Ensure that all system components and software have the latest vendor-supplied security patches installed. Install critical security patches within one month of release.” The standard goes on to state how a risk-based approach can be used to prioritize patch deployments.

As a part of the initial hardening of a system, and then throughout its lifecycle, each system should be patched fully to address known vulnerabilities. A risk-based approach is crucial: For example, a public-facing web server in the DMZ will be constantly probed and attacked. This host should arguably be patched much sooner than an intranet server that is only accessible from the core network. By the same measure, an internal server that stores critical private data will not necessarily be bright on the attack radar screen, but should be among the first to be patched.

Requirement 6.1 is followed up by 6.2 stating “Establish a process to identify newly discovered security vulnerabilities (for example, subscribe to alert services freely available on the Internet). Update configuration standards as required by PCI DSS Requirement 2.2 to address new vulnerability issues.” Note that the standard itself is relating things that you learn as a result of complying with 6.2 to actions that you are taking to be compliant with 2.2.

Related to the software development lifecycle, 6.3.1 requires “Testing of all security patches, and system and software configuration changes before deployment, including but not limited to” (cross-site scripting, injection flaws, malicious file execution, error handling, cryptographic storage, secure communications, and role-based access control).

Requirement 6.3.7 states that an organization should perform a “Review of custom code prior to release to production or customers in order to identify any potential coding vulnerability.” Additionally, 6.5 states “Develop all web applications (internal and external, and including web administrative access to application) based on secure coding guidelines such as the Open Web Application Security Project (OWASP) Guide.”

As if securing your web applications isn't difficult enough, 6.6 requires that "For public-facing web applications, address new threats and vulnerabilities on an ongoing basis and ensure these applications are protected against known attacks by either" reviewing public-facing web applications via manual or automated application vulnerability security assessment tools -or- installing a web-application firewall.

The standard gives the merchant a choice of performing a review of the application or deploying an application-layer firewall. The argument could easily be made that using these complementary approaches together would achieve the highest level of security. The intent, however, is that, in the case of a review, the merchant will also have an annual application vulnerability assessment performed. This assessment would fall short of being a penetration test, in that none of the vulnerabilities would be exploited. Corrections should be made and confirmed with re-evaluation until there is an all clear. These assessments can be performed by either an internal or external party, however, internal reviewers should be qualified to assess application-layer security, and be organizationally independent from the developers of the application.

The other choice is to protect the web applications using an application-layer firewall. These devices are available from various vendors (Barracuda, Citrix and F5, for example). These complex deployments are configured specifically for each application, and can address many of the weaknesses inherent in web applications. At a minimum, this device should protect the application from the OWASP Top Ten.

11.1, 11.2, and 11.3 require an organization to "Test for the presence of wireless access points by using a wireless analyzer at least quarterly or deploying a wireless IDS/IPS to identify all wireless devices in use", Run internal and external network vulnerability scans at least quarterly and after any significant change in the network", and perform external and internal (network-layer and application-layer) penetration testing at least annually and after any significant infrastructure or application upgrade or modification".

Internal tests for rogue wireless access points can be performed adequately by internal staff – knowledge of wireless testing tools, and some basic software and hardware is all that is required. The alternative of installing wireless IDS/IPS is a new addition in the 1.2 standard. The ultimate goal here is to discover anything connected to your network that isn't authorized, and which may present a threat to the cardholder environment.

Requirement 11.2 defines the all too familiar quarterly scanning that must be done by your Approved Scanning Vendor. 11.3 requires application-layer and network-layer penetration testing in an effort to determine whether the cardholder data can be compromised. Penetration testing takes steps beyond vulnerability testing by exploiting the vulnerabilities to determine whether a compromise of data is possible. This process may include internal and external attack vectors, packet sniffing, manual manipulation of operating systems and applications, and social engineering. The standard does not intend that tests be performed that potentially could cause a denial of service. Vulnerability scanning should be complementary to the patch management process both in terms of feeding information to patch deployment efforts (what should be patched) and as a confirmation of what has been patched.

Finding qualified penetration testers can be difficult to find (Secure Enterprise Computing provides this service) as it is often not likely that an organization will have this expertise on staff. An internal resource would have to be organizationally independent from those who manage the networking and resources.

Finally, 12.5.2 requires that you "Monitor and analyze security alerts and information, and distribute to appropriate personnel".

Security is a moving target, and staying abreast of current threats and vulnerabilities is a very important element when managing security. This applies to policy as well. Focusing on this often overlooked element can bring more benefits than a high-tech patch management system. Weak security policy should be considered a vulnerability too, and one that scanning will not discover.

## A Checklist for Vulnerability Management

So let's look at how these requirements translate into a strategy of "scheduled" events that can help keep vulnerabilities at bay and your organization in compliance with these concerns.

<b>Daily / Ongoing</b>	<ul style="list-style-type: none"><li>• Maintain configuration standards</li><li>• Monitor alert services such as SANS @Risk</li><li>• Use secure coding techniques and review code for security flaws</li></ul>
<b>Monthly</b>	<ul style="list-style-type: none"><li>• Perform vulnerability scanning of critical servers</li><li>• Apply critical patches based on risk and value</li></ul>
<b>Quarterly</b>	<ul style="list-style-type: none"><li>• Engage with an Approved Scanning Vendor to perform External Vulnerability Scans</li><li>• Scan the environment for rogue access points connected to the network</li></ul>
<b>Annually</b>	<ul style="list-style-type: none"><li>• Engage with a qualified security vendor to perform application-layer and network-layer penetration tests</li><li>• Engage with a qualified security vendor to perform application-layer vulnerability tests</li></ul>

## In Conclusion

The ongoing effort required to maintain PCI compliance requires much work and planning to be successful. Managing vulnerabilities in any environment requires more than a monthly patch routine. It requires policy, patch deployment, standardization, and testing at a minimum. An organization must intelligently prioritize and fix discovered vulnerabilities, whether by patching or other means, just to stay afloat. Hardening techniques can help reduce future vulnerabilities and prevent and protect against zero day exploits. It's very important that these technologies and efforts coordinate with one another so that the ultimate goal, protection of critical data, can be achieved.

Secure Enterprise Computing is happy to answer any questions you may have regarding the PCI Standard and how it may impact your organization. Please feel free to contact any of us at any time.



### Secure Enterprise Computing, Inc.

909 Aviation Parkway, Ste 600  
Morrisville, NC 27560

p: 919 380.7979  
f: 919 380.9055  
e: [info@secure-enterprise.com](mailto:info@secure-enterprise.com)

---

### Sales

Randall Bennett	<a href="mailto:rbennett@secure-enterprise.com">rbennett@secure-enterprise.com</a>
Karen Adkins	<a href="mailto:kadkins@secure-enterprise.com">kadkins@secure-enterprise.com</a>
Laurie Leigh	<a href="mailto:lleigh@secure-enterprise.com">lleigh@secure-enterprise.com</a>

---

### Technical

Chris Harper	<a href="mailto:charper@secure-enterprise.com">charper@secure-enterprise.com</a>
--------------	--