FortiWeb Web Application Firewall
Ensuring Compliance for PCI DSS 6.5 and 6.6
Overview

Web applications and the elements surrounding them have not only become a key part of every company’s core business infrastructure but also provide a high profile target for malicious activity. This malicious activity ranges from simple defacement attacks, denial of service attacks to more damaging data harvesting resulting in damage to customer confidence and loyalty, brand reputation and corporate credibility.

In response to major security breaches, information, identity and data theft and for the need for an environment in which consumers can engage in secure e-commerce the major five credit card companies (Visa, MasterCard, American Express, Discover and JCB) aligned their individual policies and released the Payment Card Industry Data Security Standard (PCI DSS). The standard’s intention is to create an additional level of protection for card issuers by ensuring that merchants meet minimum levels of security when they store, process and transmit cardholder data.

The Payment Card Industry Data Security Standard (PCI DSS)

The PCI DSS is a multifaceted security standard that includes requirements for security management, policies, procedures, network architecture, software design and other critical protective measures. This comprehensive standard is intended to help organizations proactively protect customer account data.

The core of the PCI DSS is a group of principles and accompanying requirements, around which the specific elements of the DSS are organized. The PCI DSS outlines 12 requirements that are put in six categories.

Build and Maintain a Secure Network

Requirement 1: Install and maintain a firewall configuration to protect cardholder data
Requirement 2: Do not use vendor-supplied defaults for system passwords and other security parameters

Protect Cardholder Data

Requirement 3: Protect stored cardholder data
Requirement 4: Encrypt transmission of cardholder data across open, public networks

Maintain a Vulnerability Management Program

Requirement 5: Use and regularly update anti-virus software
Requirement 6: Develop and maintain secure systems and applications

Implement Strong Access Control Measures

Requirement 7: Restrict access to cardholder data by business need-to-know
Requirement 8: Assign a unique ID to each person with computer access
Requirement 9: Restrict physical access to cardholder data

Regularly Monitor and Test Networks

Requirement 10: Track and monitor all access to network resources and cardholder data
Requirement 11: Regularly test security systems and processes

Maintain an Information Security Policy

Requirement 12: Maintain a policy that addresses information security

PCI Compliance for Web Applications

Out of the 12 requirements outlined in the PCI DSS web applications present some of the biggest challenges as they are usually high performance production systems which provide business driving revenue. These are usually complex applications such as e-commerce, web mail, online retail stores, social web sites, online auctions and B2B applications that quickly change, update and constantly evolve. Web applications use complex Web 2.0 technologies that create a major security challenge and as such addressing web applications as part of the PCI DSS regulation create design, policy, architecture and human resource implications.
According to the Web Application Security Consortium (WASC) 13% of web sites can be compromised completely using automatic tools. 49% of web sites contain high risk vulnerabilities prone to automatic tools while 80-96% of web applications contain high risk level that can be exploited with using hacking methods. The most common vulnerabilities are Cross Site Scripting and SQL injection.

While Fortinet provides solutions for the entire PCI DSS 12 requirements, this paper will discuss the two major requirements strategic to web applications. Section 6.5 and 6.6 mandate that by June 30, 2008 all web applications must either undergo extensive vulnerability assessments or implement a web application firewall.

- Requirement 6.5 – Develop all web applications based on secure coding guidelines to protect against Open Web Application Security Project (OWASP) Top 10.
- Requirement 6.6 - Ensure that all web-facing applications are protected against known attacks and address new threats and vulnerabilities on an ongoing basis by applying either of the following methods:
  - Having all custom application code reviewed for common vulnerabilities by an organization that specializes in application security*
  - Applications can be reviewed with manual or automated application vulnerability assessment tools or methods*
  - Installing a Web Application Firewall in front of web-facing applications.

*Applications should be reviewed at least annually and for all changes

Requirement 6.5 - Develop web applications securely based on the OWASP Top 10 guidelines
Implementing a secure coding practice as part of the development life cycle is an important part of every application development project. Web application security needs to be an essential part of any successful project. Guidelines recommended by OWASP and with requirement 6.5 help users build a more secure and trusted application, reducing the number of exploits throughout the application life cycle.

While an important role in every application development, solely relying on secure development is not enough. Challenges around this are:

- **Web 2.0 Technologies** - Modern web applications use complex Web 2.0 technologies such as Web Services, AJAX, JavaScript, Adobe Flex and others. Most developers do not have the right knowledge and expertise to be able to implement the security measures need during the relative development cycles
- **Business needs push aside needed security procedures** - Tight timeframe requirements driven by business demands cause new software updates to not go through the right secure coding practices creating new vulnerabilities with as new application version are released
- **Legacy software** – old legacy applications and/or applications inherited through mergers/acquisitions or developed by third party, out-sourced companies make it almost impossible for developers to understand and correct security flaws

Requirement 6.6 - Ensure that all web-facing applications are protected against known attacks
In order to satisfy requirement 6.6 two options are provided by the PCI standard:

- **Web application scanning** - Applications should be reviewed with manual or automated application vulnerability assessment tools to find existing vulnerabilities
- **Installing a web application firewall in front of web-facing applications** – Installing a web application firewall allows organization to block application layer attacks that might compromise credit card information

Option 1 can be achieved by conducting manual assessment or with the help of a tool. In any case, the assessments need to be performed by a qualified internal resource or a qualified third party who has the proper skills and experience to understand the web application, know how to evaluate it for vulnerabilities, and understand the findings.

The drawbacks of option 1 are that 1) some vulnerabilities are closely tied to the operating system and web/application server configurations which are typically outside of the software developer’s expertise and responsibilities; 2) to fix the broken code/application, the application/service need to be removed from live production which can introduce significant cost as well as delays in remediation exposures; 3) it is costly to acquire and maintain proper tools for vulnerability assessments, hire and train personnel capable of performing those tests; 4) it is time consuming and tedious to conduct tests that cover common vulnerability areas for each application and every time it is updated.

Achieving PCI compliance through option 2 also requires hiring and training of proper personnel, purchasing the application firewall device itself, creating and maintaining security policy. However, there are many benefits that both lower the cost of compliance and provide better level of security and accountability. Plus, it is much easier to master the management and policy development of an application firewall than perform application reviews.
However, choosing only one of these two technologies will leave a gap in the security posture of an organization and most likely create cost inefficiencies in filling those gaps. The best approach is a hybrid approach that not only scans and reports on application vulnerabilities but also protects the web application in real time with a variety of out of the box security policies.

**FortiWeb Helps Protect against OWASP Top 10**

One of the most important requirements is PCI DSS Section 6.5 as it focuses customers on the specific web application threats as defined by the Open Web Application Security Project (OWASP). The OWASP Top 10 Web Application Security Risks presents the most critical web application security flaws.

FortiWeb Web Application Firewall specifically addresses each of the OWASP Top 10 threats as portrayed in the PCI DSS 6.5 requirements.

The following is the OWASP Top 10 and the way the FortiWeb solution protects against each of the attacks:

**Injection**
Injection flaws, such as SQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query. The attacker’s hostile data can trick the interpreter into executing unintended commands or accessing unauthorized data.

**FortiWeb Web Application Firewall solution**
Auto-Learn profiling automatically creates an allowed baseline allowing a comprehensive request validation feature to enforce strict URL and parameter control. Enhanced application signature detection engine adds a secondary layer for abnormal characters and known injection strings.

**Cross-Site Scripting (XSS)**
XSS flaws occur whenever an application takes untrusted data and sends it to a web browser without proper validation and escaping. XSS allows attackers to execute scripts in the victim’s browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.

**FortiWeb Web Application Firewall solution**
Application signature detection engine includes various XSS signatures. Request validation process ensures only relevant characters can be submitted.

**Broken Authentication and Session Management**
Application functions related to authentication and session management are often not implemented correctly, allowing attackers to compromise passwords, keys, session tokens, or exploit other implementation flaws to assume other users’ identities.

**FortiWeb Web Application Firewall solution**
Enforces session management with strict cookie control. Provides various Authentication Offload capabilities (supporting Local, LDAP and NTLM) to minimize application security flaws.

**Insecure Direct Object References**
A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, or database key. Without an access control check or other protection, attackers can manipulate these references to access unauthorized data.

**FortiWeb Web Application Firewall solution**
Auto-Learn creates a comprehensive profile of allowed elements within the application. Any attempt to manipulate a parameter will trigger an alert and immediately be blocked. Hidden Fields Rules detect and block any attempt by the client to alter a hidden parameter value.

**Cross-Site Request Forgery (CSRF)**
A CSRF attack forces a logged-on victim’s browser to send a forged HTTP request, including the victim’s session cookie and any other automatically included authentication information, to a vulnerable web application. This allows the attacker to force the victim’s browser to generate requests the vulnerable application thinks are legitimate requests from the victim.

**FortiWeb Web Application Firewall solution**
Strict reference page enforcing provides protection against sophisticated CSRF attacks. Also, page access rules allow customers to define URL order. Common CSRF attacks attempt to submit a specific crafted request. For example, a payment
order. FortiWeb enforces a page order sequence that will block the request as it didn’t go through the proper payment order sequence and as such - invalid.

**Security Misconfiguration**

Good security requires having a secure configuration defined and deployed for the application, frameworks, application server, web server, database server, and platform. All these settings should be defined, implemented, and maintained as many are not shipped with secure defaults. This includes keeping all software up to date, including all code libraries used by the application.

**FortiWeb Web Application Firewall solution**

FortiWeb provides multiple ways to counter misconfigurations. Using Auto-Learn FortiWeb will block any attempt made by an attacker to exploit a misconfigured web application. Also, monitoring application responses allows FortiWeb to identify any an application failure. Lastly, FortiWeb’s unique Vulnerability Scanner feature provides the ability to scan the protected applications, find inherent misconfigurations and quickly turn these to security rules.

**Insecure Cryptographic Storage**

Many web applications do not properly protect sensitive data, such as credit cards, SSNs, and authentication credentials, with appropriate encryption or hashing. Attackers may steal or modify such weakly protected data to conduct identity theft, credit card fraud, or other crimes.

**FortiWeb Web Application Firewall solution**

Extended monitoring and protection for all outgoing traffic prevents sensitive information leakage such as credit card numbers, Social Security numbers and many other types of information disclosure.

**Failure to Restrict URL Access**

Many web applications check URL access rights before rendering protected links and buttons. However, applications need to perform similar access control checks each time these pages are accessed, or attackers will be able to forge URLs to access these hidden pages anyway.

**FortiWeb Web Application Firewall solution**

Authentication Offload allows organizations to use FortiWeb to authenticate different URLs in different Realms. Administrators can define URL groups that require specific authentication while other URLs open to the public. Using FortiWeb’s authentication capability ensures correct URL access rights are enforced.

**Insufficient Transport Layer Protection**

Applications frequently fail to authenticate, encrypt, and protect the confidentiality and integrity of sensitive network traffic. When they do, they sometimes support weak algorithms, use expired or invalid certificates, or do not use them correctly.

**FortiWeb Web Application Firewall solution**

Regardless whether the protected application uses HTTP only or weak encryption algorithms FortiWeb, when deployed as a reverse proxy adds a SSL layer and enforces SSL v3 only and cipher size higher than 128 bits to maintain strong encryption.

**Unvalidated Redirects and Forwards**

Web applications frequently redirect and forward users to other pages and websites, and use untrusted data to determine the destination pages. Without proper validation, attackers can redirect victims to phishing or malware sites, or use forwards to access unauthorized pages.

**FortiWeb Web Application Firewall solution**

Auto-Learn profiling indentifies when parameters are used in a different manner then supposed to. Validation enforcement makes sure characters that are usually associated with redirects and forwards are not allowed as part application usage.
FortiWeb Web Application Firewall Ensures Compliance with PCI 6.6

FortiWeb is the only vendor that provides a Vulnerability Scanner module within the web application firewall that completes a comprehensive solution for PCI DSS requirement 6.6 and 6.5 allowing organizations to scan their applications, find existing vulnerabilities and protect them in real time from the same platform. The following is a list of features provided by FortiWeb to help achieve PCI 6.6 compliance:

<table>
<thead>
<tr>
<th>Features</th>
<th>Benefits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto-Learn Security Profiling</td>
<td>Automatically and dynamically builds a security model of protected applications by continuously monitoring real time user activity. Access outside the baseline can then be alerted or blocked.</td>
</tr>
<tr>
<td>Application Layer Vulnerability Protection</td>
<td>Automatically scans and analyzes the protected web applications and detects security weaknesses, potential application known and unknown vulnerabilities to complete a comprehensive solution for PCI DSS</td>
</tr>
<tr>
<td>Web Application Vulnerability Assessments</td>
<td>Automatically scans and analyzes the protected web applications and detects security weaknesses, potential application known and unknown vulnerabilities to complete a comprehensive solution for PCI DSS</td>
</tr>
<tr>
<td>Data Leak Prevention</td>
<td>Extended monitoring and protection for credit card leakage and application information disclosure by tightly monitoring all outbound traffic.</td>
</tr>
<tr>
<td>Anti Web Defacement</td>
<td>Unique capabilities for monitoring protected applications for any defacement and ability to automatically and quickly revert to stored version.</td>
</tr>
<tr>
<td>Multiple Deployment Options</td>
<td>FortiWeb provides a flexible solution that can be easily introduced to any environment introduce FortiWeb in to any existing network implementations.</td>
</tr>
<tr>
<td>High Performance and Availability</td>
<td>With the integration of award winning FortiASIC technology FortiWeb is able to process tens of thousands of web transactions and provide hardware accelerated SSL offload capabilities and advanced load balancing capabilities. The high availability mode provides configuration synchronization, and allows for a network level failover in the event unexpected potential outage events.</td>
</tr>
<tr>
<td>Logging and Reporting</td>
<td>With extensive and accurate logging capability, the web application administrators can pin point the specific details when an attack happens. Hundreds of out-of-the-box report types allow administrators or auditors to analyze attacks, events, and traffics for regulatory compliance purpose</td>
</tr>
</tbody>
</table>

Mapping Fortinet Solutions to PCI Requirements

While FortiWeb helps specifically address PCI DSS requirement 6.6, Fortinet products provide a complete solution for the entire PCI DSS requirements. The following is a table outlining the requirements and the solutions Fortinet delivers to help address them:
<table>
<thead>
<tr>
<th>PCI Data Security Standard</th>
<th>Description</th>
<th>Fortinet Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build and Maintain a Secure Network</td>
<td>1. Install and maintain a firewall configuration to protect data</td>
<td>FortiGate integrated firewall functionality</td>
</tr>
<tr>
<td></td>
<td>2. Do not use vendor-supplied defaults for system passwords and other security parameters</td>
<td>FortiDB vulnerability assessment and auditing</td>
</tr>
<tr>
<td>Protect Cardholder Data</td>
<td>3. Protect stored data</td>
<td>FortiDB vulnerability assessment and monitoring</td>
</tr>
<tr>
<td></td>
<td>4. Encrypt transmission of cardholder data and sensitive information across public networks</td>
<td>FortiWeb web application security</td>
</tr>
<tr>
<td>Maintain a Vulnerability Management Program</td>
<td>5. Use and regularly update anti-virus software</td>
<td>FortiGate integrated AV</td>
</tr>
<tr>
<td></td>
<td>6. Develop and maintain secure systems and applications</td>
<td>FortiClient integrated AV</td>
</tr>
<tr>
<td></td>
<td>7. Restrict access to data by business need-to-know</td>
<td>FortiDB vulnerability assessment, auditing and monitoring</td>
</tr>
<tr>
<td></td>
<td>8. Assign a unique ID to each person with computer access</td>
<td>FortiGate integrated database or hooks to Active Directory</td>
</tr>
<tr>
<td></td>
<td>9. Restrict physical access to cardholder data</td>
<td>N/A</td>
</tr>
<tr>
<td>Implement Strong Access Control Measures</td>
<td>10. Track and monitor all access to network resources and cardholder data</td>
<td>FortiDB auditing and monitoring</td>
</tr>
<tr>
<td></td>
<td>11. Regularly test security systems and processes</td>
<td>FortiAnalyzer event reporting</td>
</tr>
<tr>
<td>Regularly Monitor and Test Networks</td>
<td>12. Maintain a policy that addresses information security</td>
<td>FortiDB vulnerability assessment</td>
</tr>
<tr>
<td>Maintain an Information Security Policy</td>
<td></td>
<td>FortiScan OS vulnerability mgmt</td>
</tr>
</tbody>
</table>
Summary

Achieving PCI DSS 6.6 compliance requires deploying a web application firewall or periodic application vulnerability scanning. FortiWeb helps customers achieve PCI DSS compliance, mitigate application attack threats and secure their business by providing a unique solution that incorporates both a web application firewall with a vulnerability assessment module.

For more information about the FortiWeb solution and other Fortinet platforms please visit www.fortinet.com