PCI For Developers
Trevor Hawthorn
Compliance will come out security, not the other way around.
Understanding PCI
Determine Scope
Card Holder Data (CHD) = Full mag strip OR PAN (Primary Account Number) (4XXX XXXX XXXX XXXX)

NO PAN = NO CHD
Pulls things into scope...

Credit Card Data

...destroys worlds
Compliant

Not Compliant
Risk
Validation
<table>
<thead>
<tr>
<th>PCI DSS Requirements</th>
<th>Testing Procedures</th>
<th>ROC Reporting Details</th>
<th>Reporting Methodology</th>
</tr>
</thead>
</table>
| **6.6 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 of the following methods:**  
- Reviewing public-facing web applications via manual or automated application vulnerability security assessment tools or methods, at least annually and after any changes  
- Installing a web-application firewall in front of public-facing web applications  | **6.6 For public-facing web applications, ensure that either one of the following methods are in place as follows:**  
- Verify that public-facing web applications are reviewed (using either manual or automated vulnerability security assessment tools or methods), as follows:  
  - At least annually  
  - After any changes  
  - By an organization that specializes in application security  
  - That all vulnerabilities are corrected  
  - That the application is re-evaluated after the corrections  
- Verify that a web-application firewall is in place in front of public-facing web applications to detect and prevent web-based attacks.  | **For each public-facing web application:**  
  i. Identify which of the two methods are implemented (web application vulnerability security assessments, web application firewalls, or both).  |  
|  | **Note:** “An organization that specializes in application security” can be either a third-party company or an internal organization, as long as the reviewers specialize in application security and can demonstrate independence from the development team.  |  |  
|  | **If application vulnerability security assessments are performed:**  
  i. Describe the tools and/or methods used (manual or automated, or a combination of both).  
  ii. Describe how it was observed that assessments are performed:  
    - At least annually  
    - After any changes  
  iii. Identify the organization(s) performing the assessments.  
  iv. Identify the responsible personnel interviewed, and describe how those reviewing the applications were confirmed to:  
    - Specialize in application security  
    - Demonstrate independence from the development team  
  v. Describe the observed process which confirm that:  
    - All identified vulnerabilities are corrected.  
    - Applications are re-evaluated after the corrections are applied.  |  |  

**PCI DSS v2.0 ROC Reporting Instructions**  
© 2011 PCI Security Standards Council LLC  
**April 2011**  
Page 59 of 112
Observe settings and configuration
Document Reviews
Personnel Interviews
Observe

Process | Action | State
Section 6.3 - SDLC

Is your SDLC mature? Is it documented?

Does the SDLC include security and PCI considerations? It must.

<table>
<thead>
<tr>
<th>Observe systems settings, config</th>
<th>Document Reviews</th>
<th>Interviews</th>
<th>Observe process, action, state</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Section 6.3.1 - Remove Custom App Accounts

// Remove non-production accounts before the app goes into production

<table>
<thead>
<tr>
<th>Observe systems settings, config</th>
<th>Document Reviews</th>
<th>Interviews</th>
<th>Observe process, action, state</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Section 6.3.2 - Code Review

// Code review policies and procedures (documentation)

// Review past code reviews and application changes (proof that it’s being done)

// You should be reviewing other developer’s code before it goes live

<table>
<thead>
<tr>
<th>Observe systems settings, config</th>
<th>Document Reviews</th>
<th>Interviews</th>
<th>Observe process, action, state</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Section 6.4 - Change Control

- Non-prod and prod separation
- Separation of duties
- Don’t use live PANs for testing
- Remove test data and non-prod accounts

<table>
<thead>
<tr>
<th>Observe systems settings, config</th>
<th>Document Reviews</th>
<th>Interviews</th>
<th>Observe process, action, state</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

// Change control procedures
// Documented impact
// Authorization
// Testing
// Back-out
Section 6.5 - Develop Secure Software

// Develop software based on secure coding guidelines

// Developer interviews

// Policy that requires training (best practices aka OWASP)

// Address the OWASP Top 10

<table>
<thead>
<tr>
<th>Observe systems settings, config</th>
<th>Document Reviews</th>
<th>Interviews</th>
<th>Observe process, action, state</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Section 6.6 - Security Testing

// Web application vulnerability assessment for *public-facing* web applications

// Manually or Automated tools

// At least annually or after “any changes”

// By firm that specializes in application security (or qualified non-conflicting internal resource)

// All vulnerabilities are fixed

// Re-tested after fixed

// Or, a web application firewall* in front of the app

<table>
<thead>
<tr>
<th>Observe systems settings, config</th>
<th>Document Reviews</th>
<th>Interviews</th>
<th>Observe process, action, state</th>
</tr>
</thead>
</table>
Case Study: Start Up

// Nov 2010 they were a Level 4 merchant

// March 2011 they were a Level 1 merchant

// Several times when they do over 1M transactions in 24 hours

// One of the top dev shops in the world

// Great security posture

// Doing almost everything right, just no policies, no formal procedures
Case Study: Airline

// Strong process and procedures

// Good policies

// Security totally not involved in development, until now

// Poor technical implementation in some areas

// Developers who thought they were out of scope, turns out were in scope
How do we get there?
// Make security part of the SDLC - document it

// Checklists reduce risk. Ask someone in the air transportation industry

// Bolt PCI on to your SDLC - document it

// Train your developers to write secure code. Ask Sony about this.
Or... dodge the bullet

// Explore outsourcing payment process to 3rd party
// <iframe>
// billing.companyname.com - DNS
// Technical and branding limitations
Innovative Risk Solutions

trevor.hawthorn@stratumsecurity.com

www.stratumsecurity.com